Table of Contents :

Contents:

(Problem 1)

1.1 Read the dataset. Do the descriptive statistics and do the null value condition check. Write an inference on it. (4 Marks)

1.2 Perform Univariate and Bivariate Analysis. Do exploratory data analysis. Check for Outliers. (7 Marks)

Data Preparation: 4 marks
1.3 Encode the data (having string values) for Modelling. Is Scaling necessary here or not? Data Split: Split the data into train and test (70:30). (4 Marks)
Modeling: 22 marks

1.4 Apply Logistic Regression and LDA (linear discriminant analysis). (4 marks)

1.5 Apply KNN Model and Naïve Bayes Model. Interpret the results. (4 marks)

1.6 Model Tuning, Bagging (Random Forest should be applied for Bagging), and Boosting. (7 marks)

1.7 Performance Metrics: Check the performance of Predictions on Train and Test sets using Accuracy, Confusion Matrix, Plot ROC curve and get ROC_AUC score for each model. Final Model: Compare the models and write inference which model is best/optimized. (7 marks)

Inference: 5 marks
1.8 Based on these predictions, what are the insights? (5 marks)

(Problem 2)

In this particular project, we are going to work on the inaugural corpora from the nltk in Python. We will be looking at the following speeches of the Presidents of the United States of America:

1. President Franklin D. Roosevelt in 1941

2. President John F. Kennedy in 1961

3. President Richard Nixon in 1973

(Hint: use .words(), .raw(), .sent() for extracting counts)

2.1 Find the number of characters, words, and sentences for the mentioned documents. – 3 Marks

2.2 Remove all the stopwords from all three speeches. – 3 Marks

2.3 Which word occurs the most number of times in his inaugural address for each president?

Mention the top three words. (after removing the stopwords) – 3 Marks
2.4 Plot the word cloud of each of the speeches of the variable. (after removing the stopwords) – 3 Marks [refer to the End-to-End Case Study done in the Mentored Learning Session]

Problem 1:

You are hired by one of the leading news channels CNBE who wants to analyze recent elections. This survey was conducted on 1525 voters with 9 variables. You have to build a model, to predict which party a voter will vote for on the basis of the given information, to create an exit poll that will help in predicting overall win and seats covered by a particular party.

1.1 Read the dataset. Do the descriptive statistics and do the null value condition check. Write an inference on it.

Dataset :

	Unnamed: 0	vote	age	economic.cond.national	economic.cond.household	Blair	Hague	Europe	political.knowledge	gender
0	1	Labour	43	3	3	4	1	2	2	female
1	2	Labour	36	4	4	4	4	5	2	male
2	3	Labour	35	4	4	5	2	3	2	male
3	4	Labour	24	4	2	2	1	4	0	female
4	5	Labour	41	2	2	1	1	6	2	male

Removing Unnamed Column:

	vote	age	economic.cond.national	economic.cond.household	Blair	Hague	Europe	political.knowledge	gender
0	Labour	43	3	3	4	1	2	2	female
1	Labour	36	4	4	4	4	5	2	male
2	Labour	35	4	4	5	2	3	2	male

	vote	age	economic.cond.national	economic.cond.household	Blair	Hague	Europe	political.knowledge	gender
3	Labour	24	4	2	2	1	4	0	female
4	Labour	41	2	2	1	1	6	2	male

Shape:

```
Number of rows:  1525
Number. of columns:  9
```

Info:

```
#   Column                   Non-Null Count  Dtype
---  ------                   --------------  -----
 0   vote                     1525 non-null   object
 1   age                      1525 non-null   int64
 2   economic.cond.national   1525 non-null   int64
 3   economic.cond.household  1525 non-null   int64
 4   Blair                    1525 non-null   int64
 5   Hague                    1525 non-null   int64
 6   Europe                   1525 non-null   int64
 7   political.knowledge      1525 non-null   int64
 8   gender                   1525 non-null   object
```

Dtypes:

```
vote                         object
age                           int64
economic.cond.national        int64
economic.cond.household       int64
Blair                         int64
Hague                         int64
Europe                        int64
political.knowledge           int64
gender                       object
```

Null Values Check:

```
vote                       0
age                        0
economic.cond.national     0
economic.cond.household    0
Blair                      0
Hague                      0
Europe                     0
political.knowledge        0
gender                     0
```

Duplicates :

```
Number of duplicate rows = 8
```

	vote	age	economic.cond.national	economic.cond.household	Blair	Hague	Europe	political.knowledge	gender
67	Labour	35	4	4	5	2	3	2	male
626	Labour	39	3	4	4	2	5	2	male
870	Labour	38	2	4	2	2	4	3	male
983	Conservative	74	4	3	2	4	8	2	female
1154	Conservative	53	3	4	2	2	6	0	female
1236	Labour	36	3	3	2	2	6	2	female
1244	Labour	29	4	4	4	2	2	2	female
1438	Labour	40	4	3	4	2	2	2	male

Describe :

	age	economic.cond.national	economic.cond.household	Blair	Hague	Europe	political.knowledge
count	1517.000000	1517.000000	1517.000000	1517.000000	1517.000000	1517.000000	1517.000000
mean	54.241266	3.245221	3.137772	3.335531	2.749506	6.740277	1.540541
std	15.701741	0.881792	0.931069	1.174772	1.232479	3.299043	1.084417
min	24.000000	1.000000	1.000000	1.000000	1.000000	1.000000	0.000000
25%	41.000000	3.000000	3.000000	2.000000	2.000000	4.000000	0.000000
50%	53.000000	3.000000	3.000000	4.000000	2.000000	6.000000	2.000000
75%	67.000000	4.000000	4.000000	4.000000	4.000000	10.000000	2.000000
max	93.000000	5.000000	5.000000	5.000000	5.000000	11.000000	3.000000

Categorical variable :

	vote	gender
count	1517	1517
unique	2	2
top	Labour	female
freq	1057	808

Value Counts :

```
AGE :  70
91     1
93     1
90     1
92     2
87     3
      ..
46    37
47    38
35    38
49    39
37    42
Name: age, Length: 70, dtype: int64


ECONOMIC.COND.NATIONAL :  5
1     37
5     82
2    256
4    538
3    604
Name: economic.cond.national, dtype: int64


ECONOMIC.COND.HOUSEHOLD :  5
1     65
5     92
2    280
4    435
3    645
Name: economic.cond.household, dtype: int64


BLAIR :  5
3      1
1     97
5    152
2    434
4    833
Name: Blair, dtype: int64


HAGUE :  5
3     37
5     73
1    233
4    557
2    617
Name: Hague, dtype: int64


EUROPE :  11
2      77
7      86
10    101
1     109
9     111
8     111
```

```
5     123
4     126
3     128
6     207
11    338
Name: Europe, dtype: int64


POLITICAL.KNOWLEDGE :  4
1     38
3    249
0    454
2    776
Name: political.knowledge, dtype: int64
```

Categorical Variable :

```
VOTE :  2
Conservative     460
Labour          1057
Name: vote, dtype: int64


GENDER :  2
male      709
female    808
Name: gender, dtype: int64
```

Skewness:

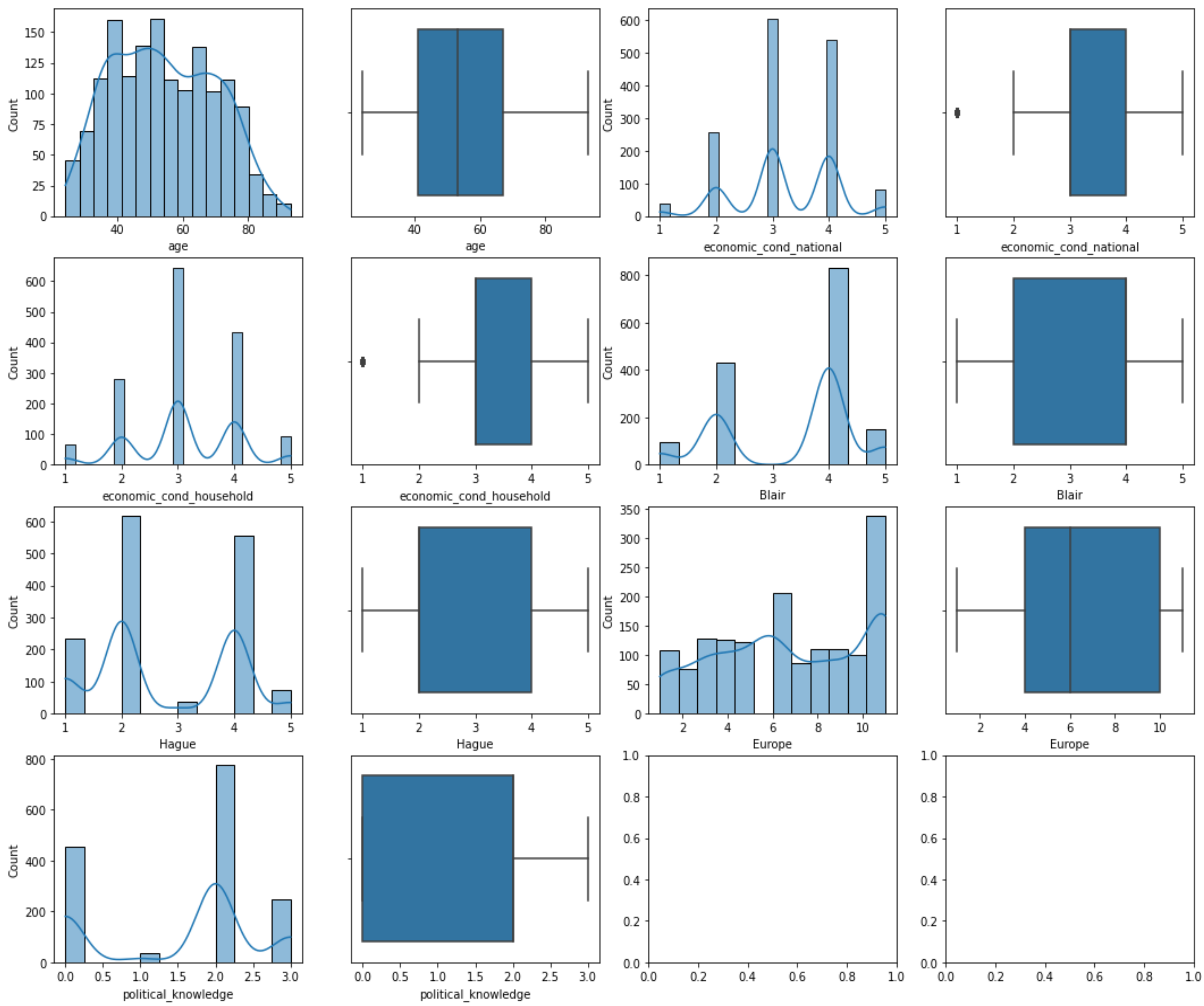
```
Skewness values
age                         0.139800
economic.cond.national     -0.238474
economic.cond.household    -0.144148
Blair                      -0.539514
Hague                       0.146191
Europe                     -0.141891
political.knowledge        -0.422928
```

NA Values check:

```
vote                       0
age                        0
economic.cond.national     0
economic.cond.household    0
Blair                      0
Hague                      0
Europe                     0
political.knowledge        0
gender                     0
```

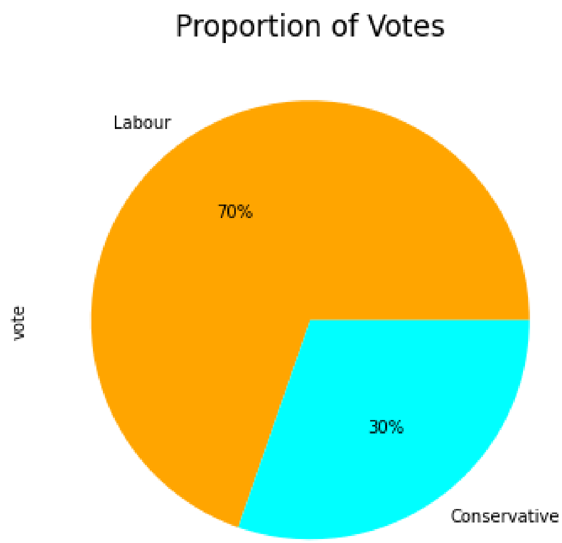
1.2 Perform Univariate and Bivariate Analysis. Do exploratory data analysis. Check for Outliers.

Box & Histplots:



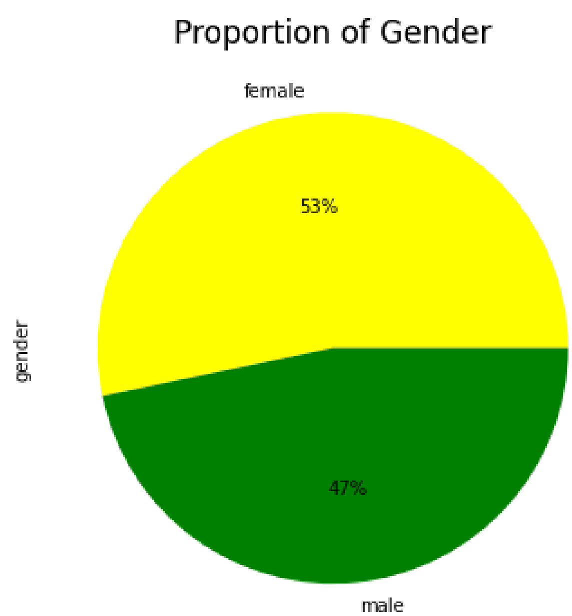
Here, we have two variables with outliers but since these are which are numeric but present us a position which is from 1 to 5. 1 is for worst and 5 is for excellent. These rating gives us a idea of the economic condition of both national and household income so we are not going to change or treat it otherwise data will be lost, and losing data is not good for us as it is a election pole we need as much data as we can so that we get good accuracy.

Proportion of Votes:



Here, we can see proportion of votes by our two clases in our target variable.

Proportion of Gender:



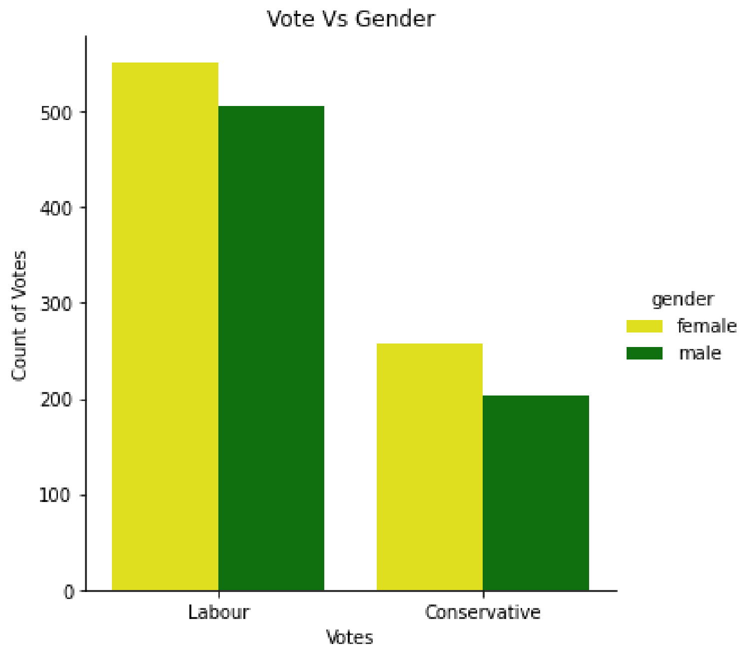
Here, we can see total proportion of Male and female who voted in this election.

Male : 47%

Female : 53%

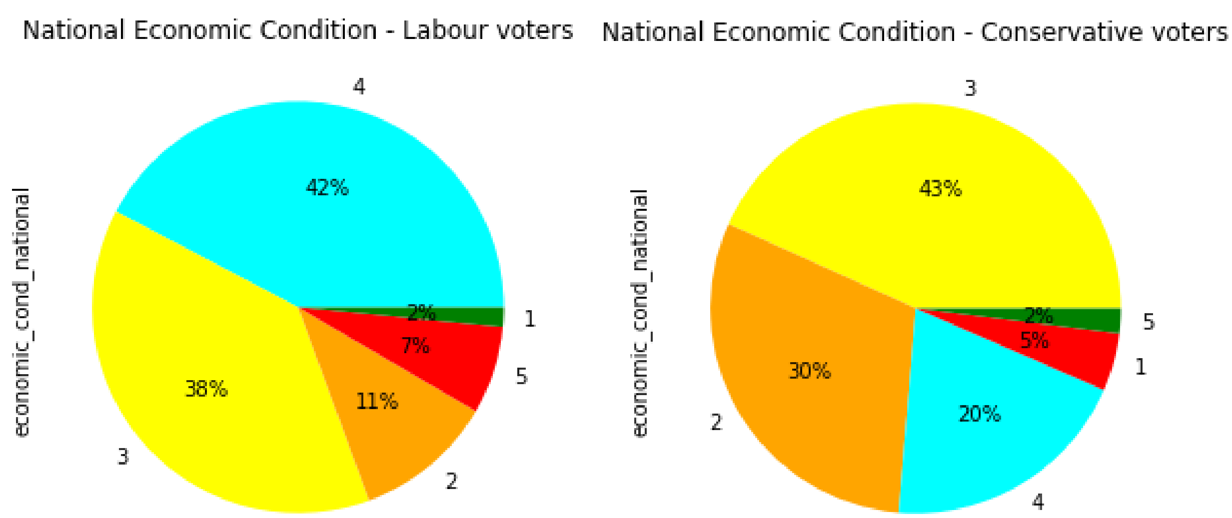
Female population is more than male one.

Vote Vs Gender:



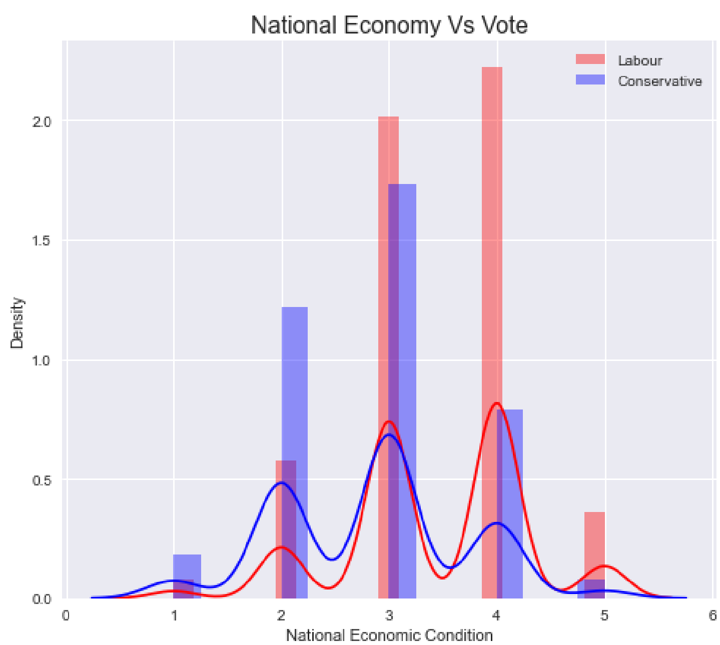
Here, we can see whose count of vote is more in the given two clases according to the gender. So, here we can see female are more in in both the variable and most of the population is coming or have given vote are from labour class.

National Economic Condition - Labour voters & National Economic Condition - Conservative voters



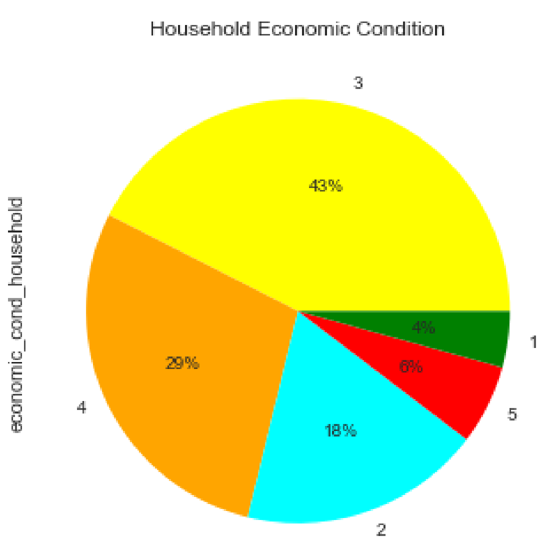
Here, we can see the proportion of votes coming from the target variable according to the given ratings in our National Economic Condition variable here we can see that from labour class which have a rating of 4 has the highest proportion of voters coming from there and in Conservative class it's 43% which have a rating of 3.These rating tell us about the economic condition. 1 = Worst, 5 = Excellent economic condition.

National Economy Vs Vote



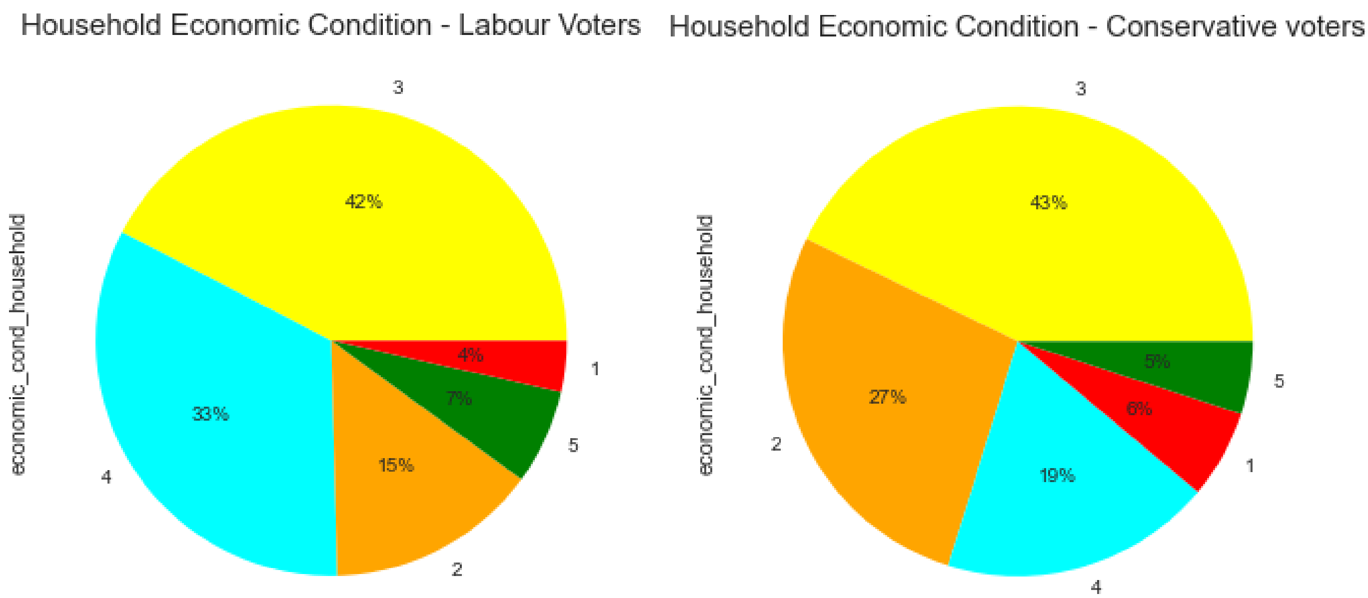
Here, we can see the votes from the given classes with respect to national economic condition.

Household Economic Condition



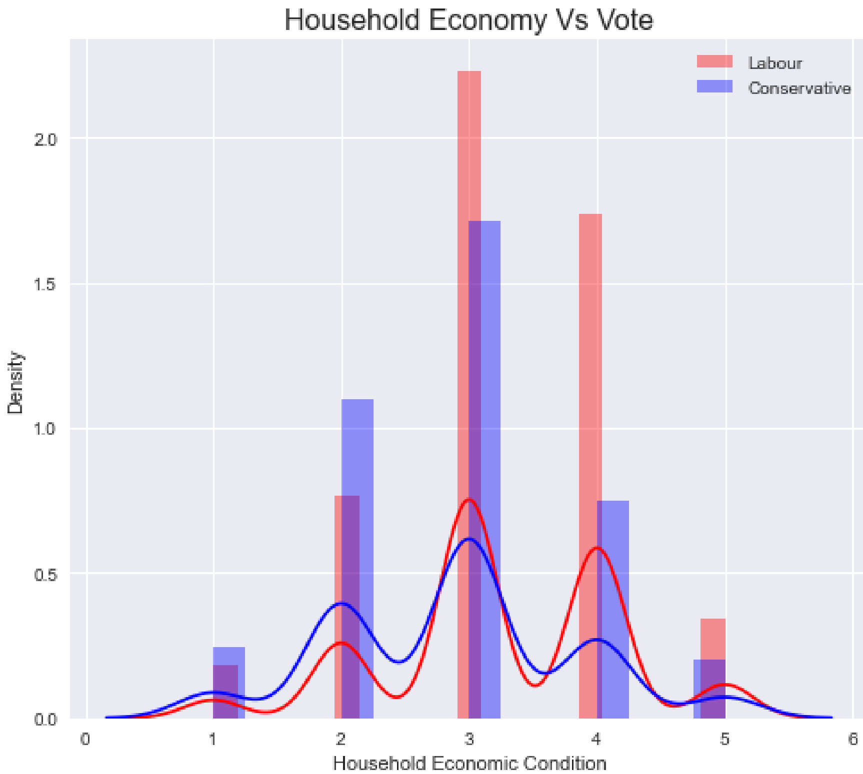
Here, we can see household economic conditions with respect to ratings.

Household Economic Condition - Labour Voters & Household Economic Condition - Conservative voters



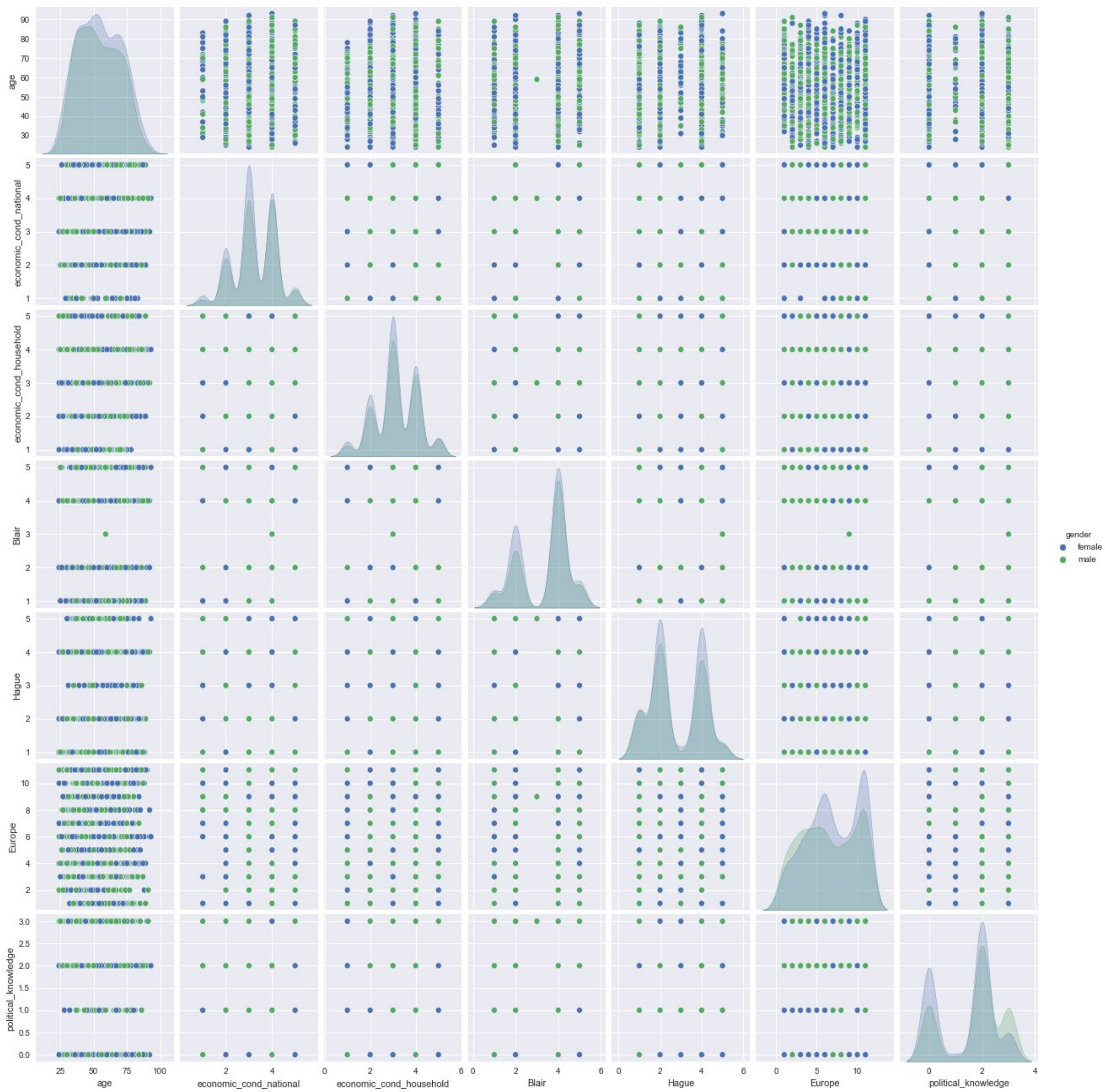
Here, we can see the proportion of votes coming from the target variable according to the given ratings in our Household Economic Condition variable here we can see that from labour class which have a rating of 3 has the highest proportion of voters coming from there and in Conservative class it’s 43% which have a rating of 3.These rating tell us about the economic condition. 1 = Worst, 5 = Excellent economic condition.

Household Economy Vs Vote



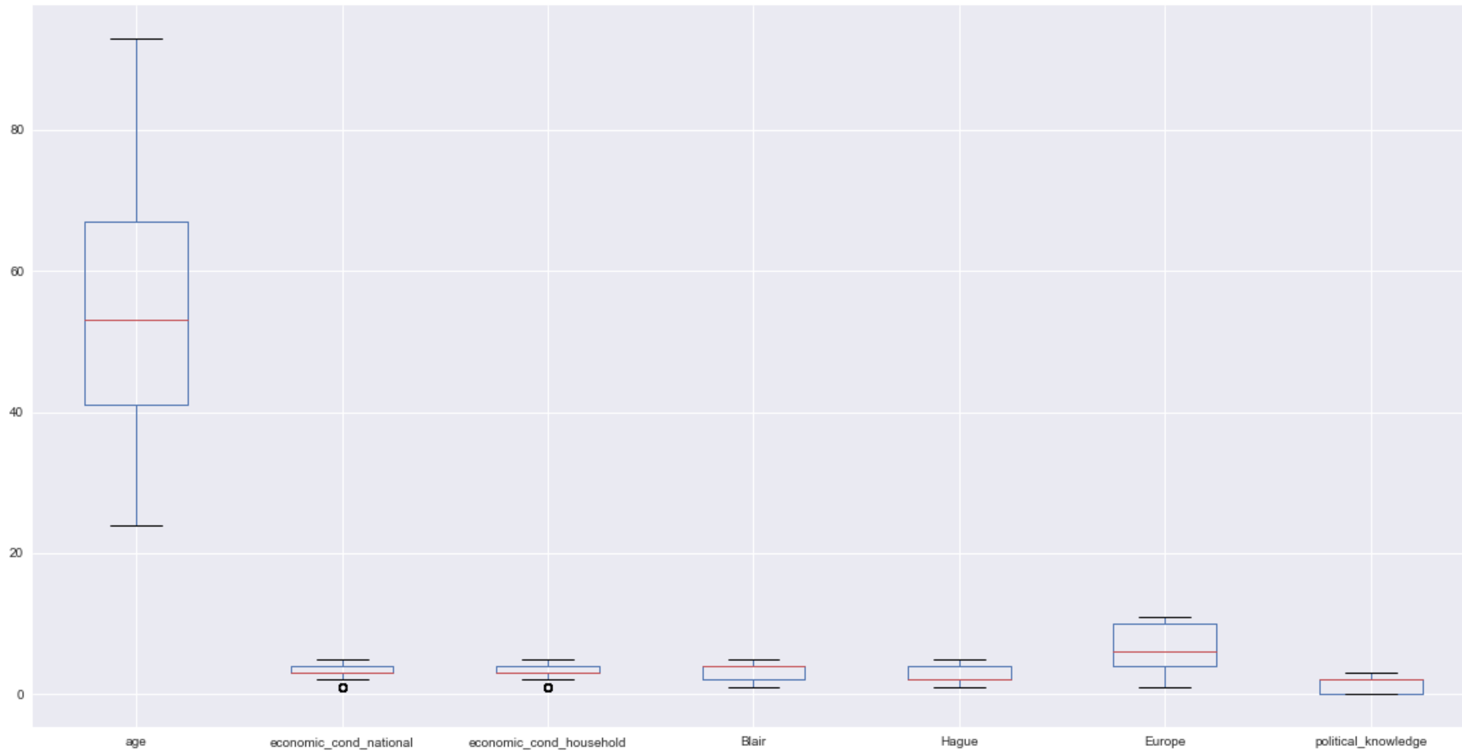
Here, we can see the votes from the given classes with respect to Household economic condition. Labour class having the most number of proportion as well as votes too in comparision to conservative class.

Pairplot:

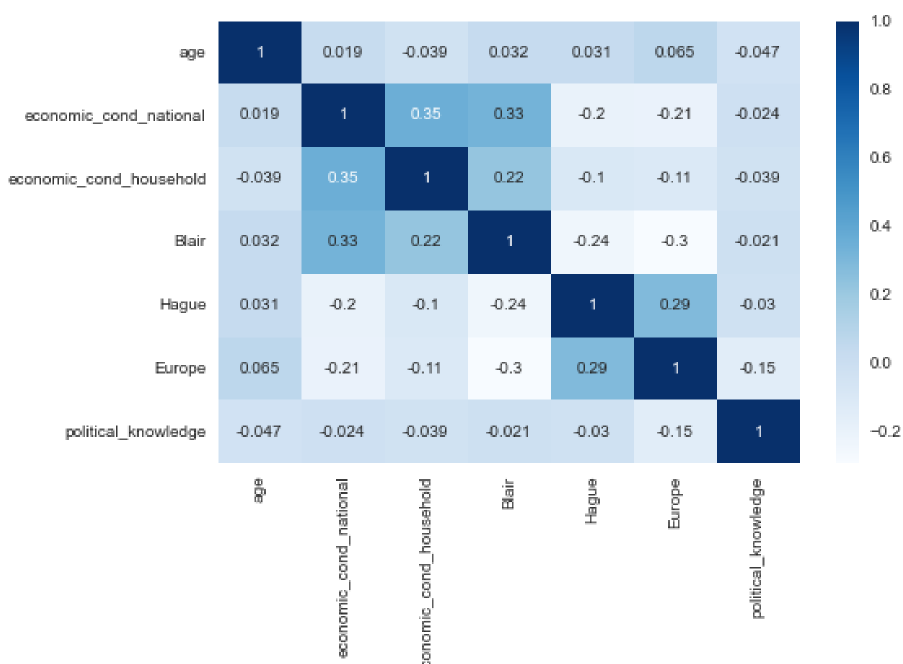


we doesnt have much correlation between variables.

Outlier:



Heatmap:



Collinearity is very low.

1.3 Encode the data (having string values) for Modelling. Is Scaling necessary here or not? Data Split: Split the data into train and test (70:30).

Dividing Category into Categorical and Numerical:

```
['vote', 'gender']
['age', 'economic_cond_national', 'economic_cond_household', 'Bl
air', 'Hague', 'Europe', 'political_knowledge']
```

Table after renaming:

	age	economic_cond_national	economic_cond_household	Blair	Hague	Europe	political_knowledge	Conservative _Labour	Male_Female
0	43	3	3	4	1	2	2	1	0
1	36	4	4	4	4	5	2	1	1
2	35	4	4	5	2	3	2	1	1
3	24	4	2	2	1	4	0	1	0
4	41	2	2	1	1	6	2	1	1

Changed Dtype of the 2 categorical variable:

```
#   Column                   Non-Null Count  Dtype
---  ------                   --------------  -----
 0   age                      1517 non-null   int64
 1   economic_cond_national   1517 non-null   int64
 2   economic_cond_household  1517 non-null   int64
 3   Blair                    1517 non-null   int64
 4   Hague                    1517 non-null   int64
 5   Europe                   1517 non-null   int64
 6   political_knowledge      1517 non-null   int64
 7   Conservative _Labour     1517 non-null   int64
 8   Male_Female              1517 non-null   int64
```

1.4 Apply Logistic Regression and LDA (linear discriminant analysis). Splitted the data

LinearDiscriminantAnalysis:

Train:

```
0.8341187558906692
[[200 107]
 [ 69 685]]
              precision    recall  f1-score   support

           0       0.74      0.65      0.69       307
           1       0.86      0.91      0.89       754

    accuracy                           0.83      1061
   macro avg       0.80      0.78      0.79      1061
weighted avg       0.83      0.83      0.83      1061
```

Test:

```
0.8333333333333334
[[111  42]
 [ 34 269]]
              precision    recall  f1-score   support

           0       0.77      0.73      0.74       153
           1       0.86      0.89      0.88       303

    accuracy                           0.83       456
   macro avg       0.82      0.81      0.81       456
weighted avg       0.83      0.83      0.83       456
```

LogisticRegression

Train :

```
0.8350612629594723
[[199 108]
 [ 67 687]]
              precision    recall  f1-score   support

           0       0.75      0.65      0.69       307
           1       0.86      0.91      0.89       754

    accuracy                           0.84      1061
   macro avg       0.81      0.78      0.79      1061
weighted avg       0.83      0.84      0.83      1061
```

Test :

```
0.8245614035087719
[[110  43]
 [ 37 266]]
              precision    recall  f1-score   support

           0       0.75      0.72      0.73       153
           1       0.86      0.88      0.87       303

    accuracy                           0.82       456
   macro avg       0.80      0.80      0.80       456
weighted avg       0.82      0.82      0.82       456
```

1.5 Apply KNN Model and Naïve Bayes Model. Interpret the results.

Split data : (Applied Zscore)

Table :

	age	economic_cond_national	economic_cond_household	Blair	Hague	Europe	political_knowledge	Male_Female
0	-0.716161	-0.278185	-0.148020	0.565802	-1.419969	-1.437338	0.423832	-0.936736
1	-1.162118	0.856242	0.926367	0.565802	1.014951	-0.527684	0.423832	1.067536
2	-1.225827	0.856242	0.926367	1.417312	-0.608329	-1.134120	0.423832	1.067536
3	-1.926617	0.856242	-1.222408	-1.137217	-1.419969	-0.830902	-1.421084	-0.936736
4	-0.843577	-1.412613	-1.222408	-1.988727	-1.419969	-0.224465	0.423832	1.067536

Train:

```
0.8557964184731386
[[218  89]
 [ 64 690]]
              precision    recall  f1-score   support

           0       0.77      0.71      0.74       307
           1       0.89      0.92      0.90       754

    accuracy                           0.86      1061
   macro avg       0.83      0.81      0.82      1061
weighted avg       0.85      0.86      0.85      1061
```

Test :

```
0.8245614035087719
[[105  48]
 [ 32 271]]
              precision    recall  f1-score   support

           0       0.77      0.69      0.72       153
           1       0.85      0.89      0.87       303

    accuracy                           0.82       456
```

```
   macro avg       0.81      0.79      0.80       456
weighted avg       0.82      0.82      0.82       456
```

Naive Bayes Model

Train:

```
0.8350612629594723
[[211  96]
 [ 79 675]]
              precision    recall  f1-score   support

           0       0.73      0.69      0.71       307
           1       0.88      0.90      0.89       754

    accuracy                           0.84      1061
   macro avg       0.80      0.79      0.80      1061
weighted avg       0.83      0.84      0.83      1061
```

Test :

```
0.8223684210526315
[[112  41]
 [ 40 263]]
              precision    recall  f1-score   support

           0       0.74      0.73      0.73       153
           1       0.87      0.87      0.87       303

    accuracy                           0.82       456
   macro avg       0.80      0.80      0.80       456
weighted avg       0.82      0.82      0.82       456
```

1.6 Model Tuning, Bagging (Random Forest should be applied for Bagging), and Boosting.

Value Count of our target variable:

```
1    1057
0     460
```

Model Tuning

Linear Regression with SMOTE

Train:

	precision	recall	f1-score	support
0	0.828794	0.847480	0.838033	754.000000
1	0.843962	0.824934	0.834339	754.000000
accuracy	0.836207	0.836207	0.836207	0.836207
macro avg	0.836378	0.836207	0.836186	1508.000000
weighted avg	0.836378	0.836207	0.836186	1508.000000

Test:

	precision	recall	f1-score	support
0	0.664865	0.803922	0.727811	153.000000
1	0.889299	0.795380	0.839721	303.000000
accuracy	0.798246	0.798246	0.798246	0.798246
macro avg	0.777082	0.799651	0.783766	456.000000
weighted avg	0.813995	0.798246	0.802172	456.000000

LDA with SMOTE

Train :

	precision	recall	f1-score	support
0	0.829237	0.850133	0.839555	754.000000
1	0.846259	0.824934	0.835460	754.000000
accuracy	0.837533	0.837533	0.837533	0.837533
macro avg	0.837748	0.837533	0.837507	1508.000000
weighted avg	0.837748	0.837533	0.837507	1508.000000

Test:

	precision	recall	f1-score	support
0	0.666667	0.823529	0.736842	153.000000
1	0.898876	0.792079	0.842105	303.000000
accuracy	0.802632	0.802632	0.802632	0.802632
macro avg	0.782772	0.807804	0.789474	456.000000
weighted avg	0.820964	0.802632	0.806787	456.000000

KNN with SMOTE

Train:

	precision	recall	f1-score	support
0	0.838973	0.953581	0.892613	754.000000
1	0.946237	0.816976	0.876868	754.000000
accuracy	0.885279	0.885279	0.885279	0.885279
macro avg	0.892605	0.885279	0.884741	1508.000000
weighted avg	0.892605	0.885279	0.884741	1508.000000

Test :

	precision	recall	f1-score	support
0	0.672131	0.803922	0.732143	153.000000
1	0.890110	0.801980	0.843750	303.000000
accuracy	0.802632	0.802632	0.802632	0.802632
macro avg	0.781121	0.802951	0.787946	456.000000
weighted avg	0.816972	0.802632	0.806303	456.000000

Naive Bayes

Train:

	precision	recall	f1-score	support
0	0.832891	0.832891	0.832891	754.000000
1	0.832891	0.832891	0.832891	754.000000
accuracy	0.832891	0.832891	0.832891	0.832891
macro avg	0.832891	0.832891	0.832891	1508.000000
weighted avg	0.832891	0.832891	0.832891	1508.000000

Test:

	precision	recall	f1-score	support
0	0.687861	0.777778	0.730061	153.000000
1	0.879859	0.821782	0.849829	303.000000
accuracy	0.807018	0.807018	0.807018	0.807018
macro avg	0.783860	0.799780	0.789945	456.000000
weighted avg	0.815438	0.807018	0.809644	456.000000

Hyperparameter tuning using GridsearchCV

Logistic Regression with GridSearchCV

```
GridSearchCV(cv=10, estimator=LogisticRegression(class_weight={0: 2, 1: 1}),
             n_jobs=-1,
             param_grid={'C': array([1.00000000e-03, 2.06913808e-03, 4.28133240e-03,
8.85866790e-03,
       1.83298071e-02, 3.79269019e-02, 7.84759970e-02, 1.62377674e-01,
       3.35981829e-01, 6.95192796e-01, 1.43844989e+00, 2.97635144e+00,
       6.15848211e+00, 1.27427499e+01, 2.63665090e+01, 5.45559478e+01,
       1.12883789e+02, 2.33572147e+02, 4.83293024e+02, 1.00000000e+03]),
                         'penalty': ['l2', 'none'],
                         'solver': ['newton-cg', 'lbfgs', 'sag', 'saga']})
```

estimator: LogisticRegression

```
LogisticRegression(class_weight={0: 2, 1: 1})
```

```
LogisticRegression
```

```
LogisticRegression(class_weight={0: 2, 1: 1})
```

Best Parametres

```
Best Parametres from LogisticRegression(C=0.0379269019073225, cl
ass_weight={0: 2, 1: 1},
                      solver='newton-cg')
```

Logistic regression does not really have any critical hyperparameters to tune.

Sometimes, you can see useful differences in performance or convergence with different solvers (solver).

solver in [‘newton-cg’, ‘lbfgs’, ‘liblinear’, ‘sag’, ‘saga’] Regularization (penalty) can sometimes be helpful.

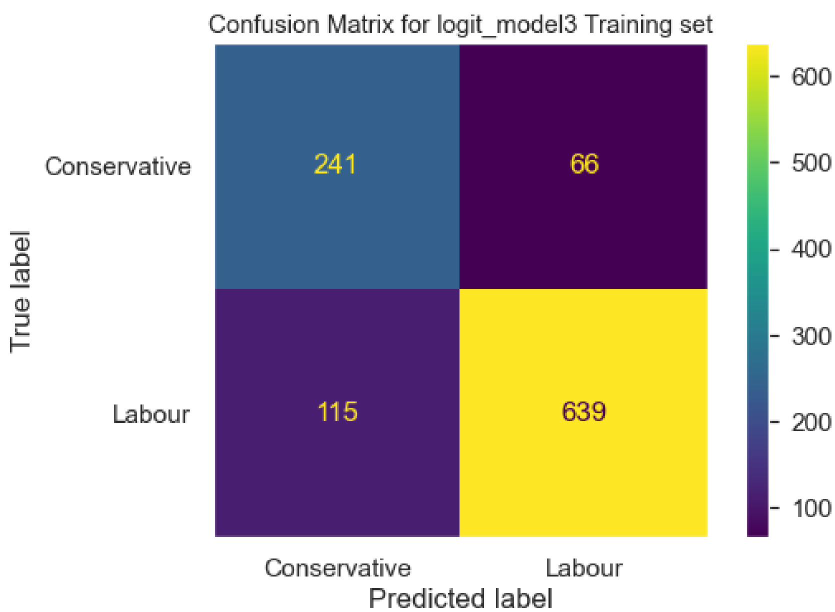
penalty in [‘none’, ‘l1’, ‘l2’, ‘elasticnet’] Note: not all solvers support all regularization terms.

The C parameter controls the penality strength, which can also be effective.

C in [100, 10, 1.0, 0.1, 0.01]

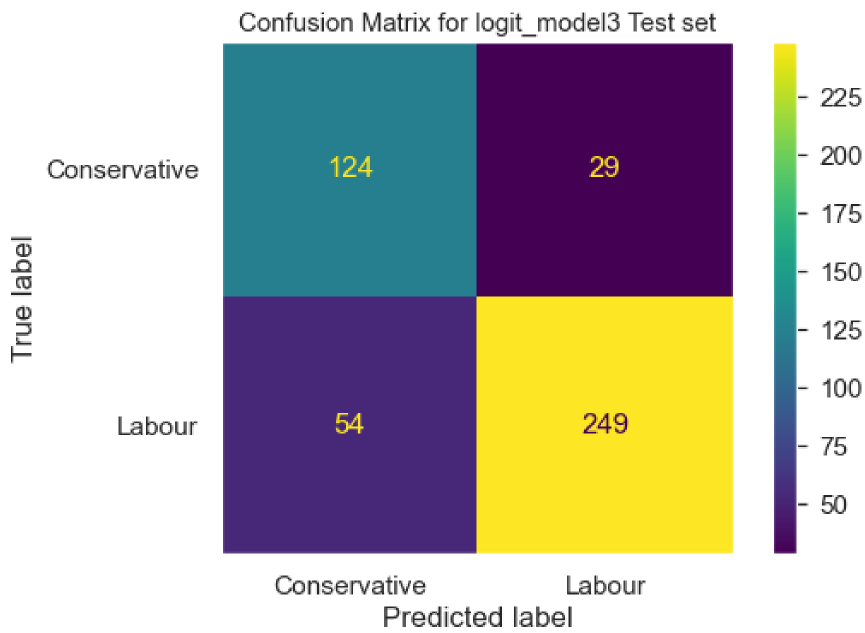
Train :

	precision	recall	f1-score	support
0	0.68	0.79	0.73	307
1	0.91	0.85	0.88	754
accuracy			0.83	1061
macro avg	0.79	0.82	0.80	1061
weighted avg	0.84	0.83	0.83	1061

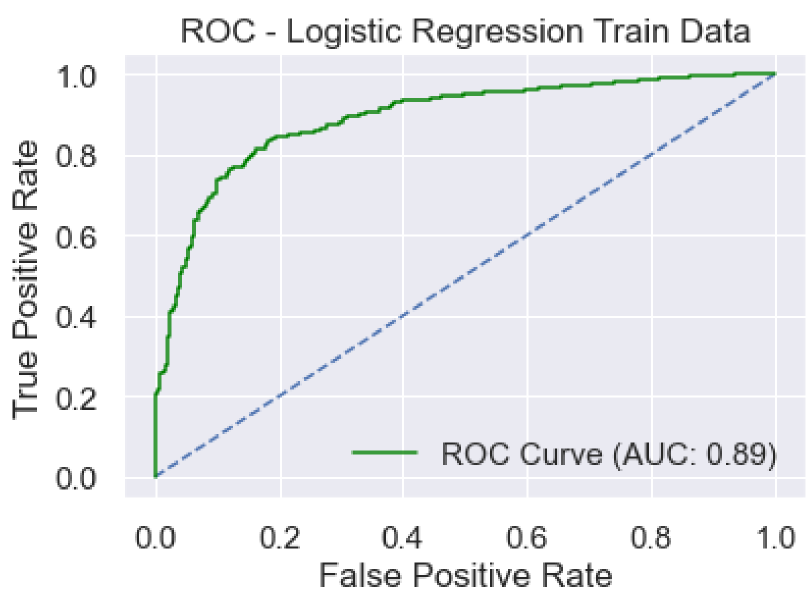


Test :

	precision	recall	f1-score	support
0	0.70	0.81	0.75	153
1	0.90	0.82	0.86	303
accuracy			0.82	456
macro avg	0.80	0.82	0.80	456
weighted avg	0.83	0.82	0.82	456



ROC Curve Train:



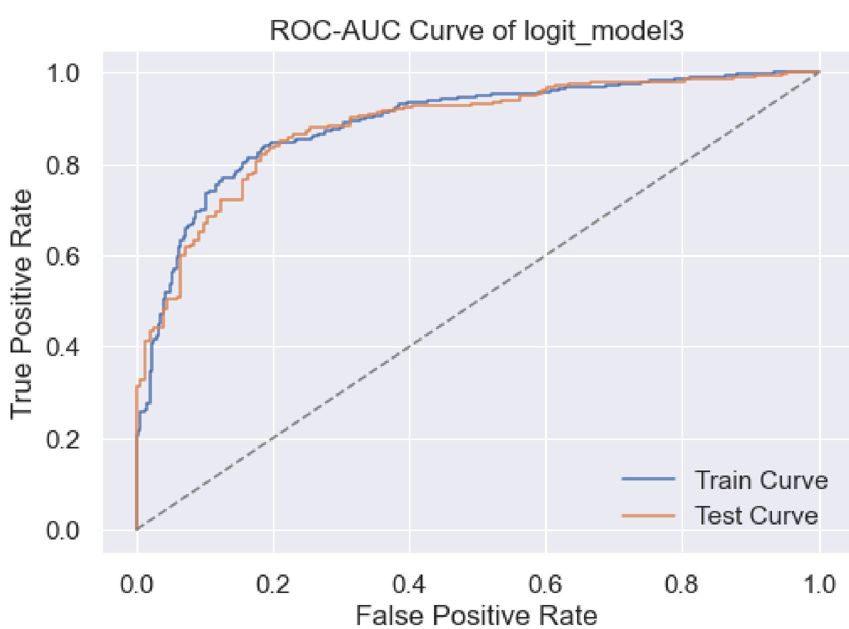
logit_train_auc 0.8897130612844417

ROC Curve Test:

logit_test_auc 0.8836471882482366

Combined :

AUC for Training data = 0.8897130612844417
AUC for Test data = 0.8836471882482366



Linear Discriminant Analysis with GridsearchCV

```
GridSearchCV(cv=3, estimator=LinearDiscriminantAnalysis(),
             param_grid={'solver': ['svd', 'lsqr', 'eigen'],
                         'tol': [0.0001, 0.001, 0.01]})
```

estimator: LinearDiscriminantAnalysis

```
LinearDiscriminantAnalysis()
```

```
LinearDiscriminantAnalysis
```

```
LinearDiscriminantAnalysis()
```

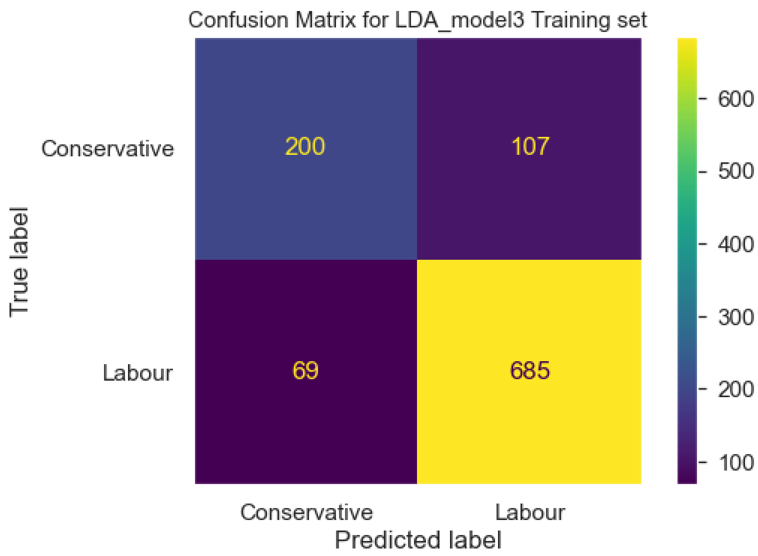
Best Parameters from LDA {'solver': 'svd', 'tol' : 0.0001}

Sometimes, you can see useful differences in performance or convergence with different solvers (solver).

tol: Absolute threshold for a singular value of X to be considered significant, used to estimate the rank of X. Dimensions whose singular values are non-significant are discarded. Only used if solver is ‘svd’.

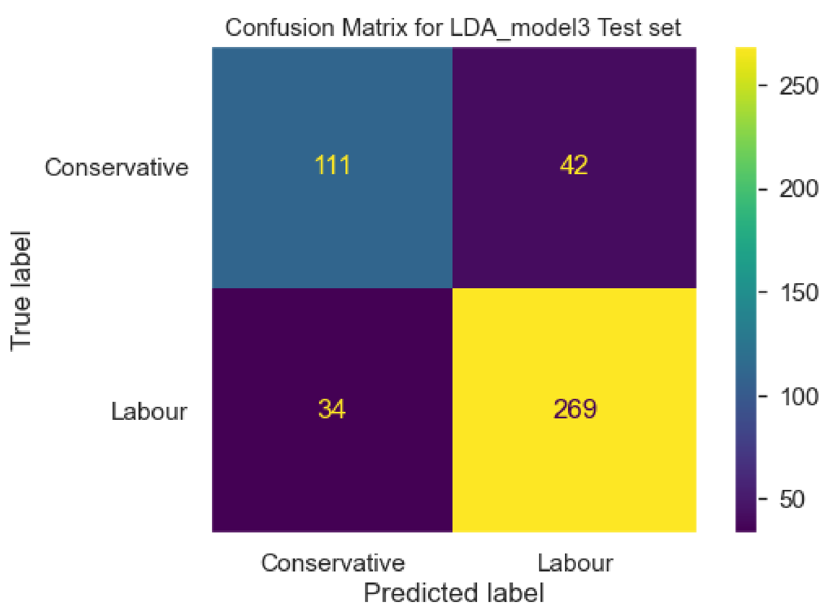
Train :

	precision	recall	f1-score	support
0	0.74	0.65	0.69	307
1	0.86	0.91	0.89	754
accuracy			0.83	1061
macro avg	0.80	0.78	0.79	1061
weighted avg	0.83	0.83	0.83	1061

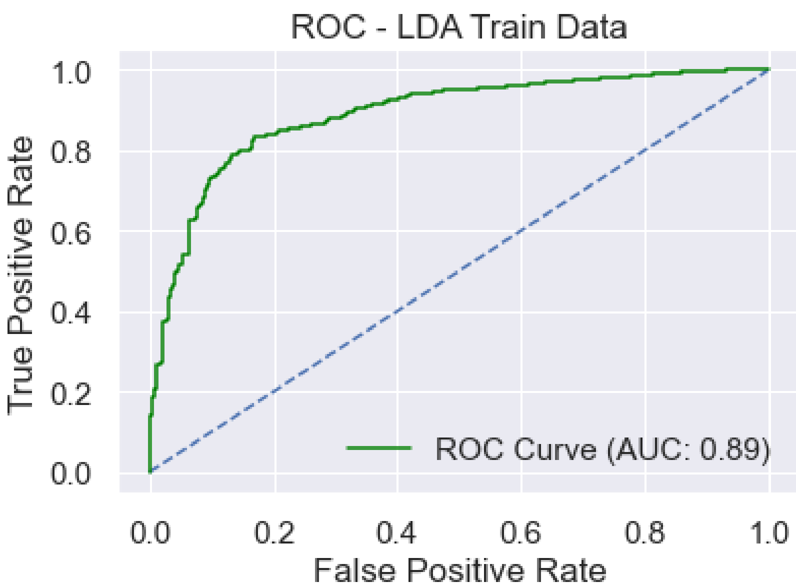


Test :

	precision	recall	f1-score	support
0	0.77	0.73	0.74	153
1	0.86	0.89	0.88	303
accuracy			0.83	456
macro avg	0.82	0.81	0.81	456
weighted avg	0.83	0.83	0.83	456

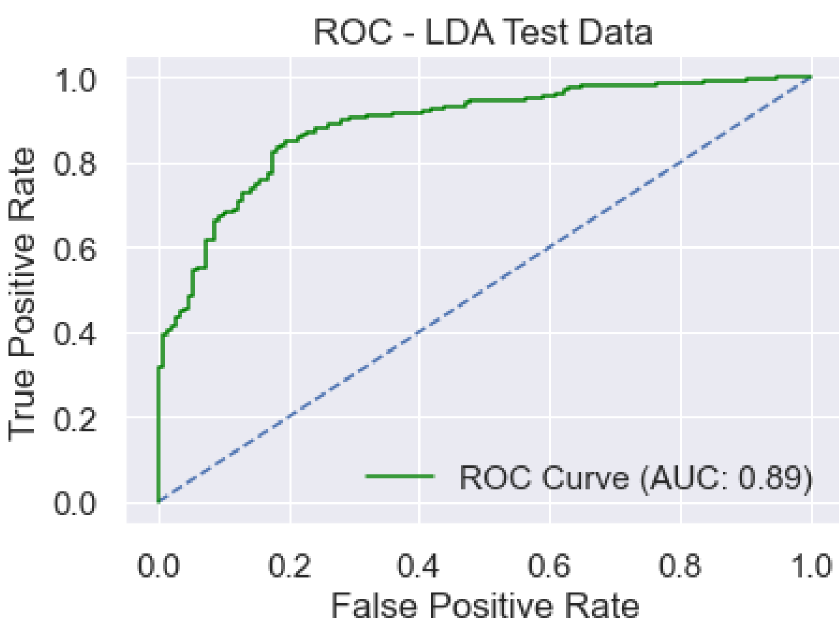


ROC TRAIN DATA :



LDA_train_auc 0.8893674560865394

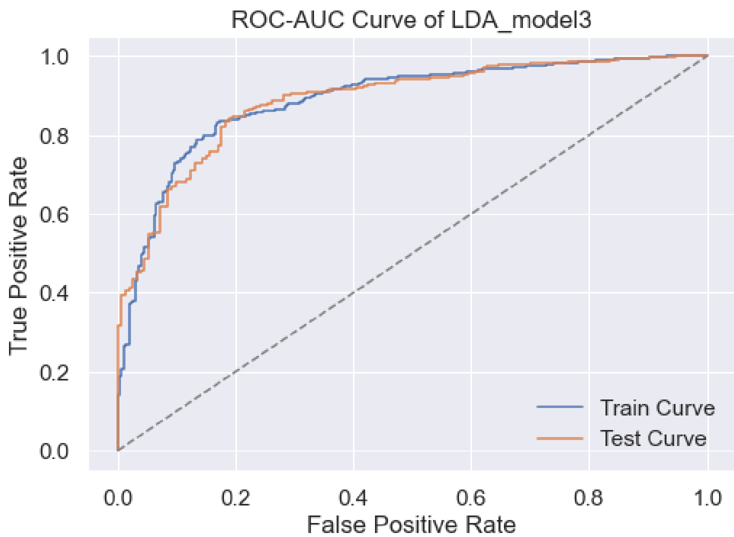
ROC teat data :



LDA_test_auc 0.8876377833861817

Combined :

AUC for Training data = 0.8893674560865394
AUC for Test data = 0.8876377833861817



KNN Model with GridsearchCV

```
GridSearchCV(cv=5, estimator=KNeighborsClassifier(),
             param_grid={'metric': ['minkowski', 'euclidean', 'canberra'],
                         'n_neighbors': range(5, 20), 'weights': ['uniform
']})
```

estimator: KNeighborsClassifier

```
KNeighborsClassifier()
```

KNeighborsClassifier

```
KNeighborsClassifier()
```

Best Parameters from KNN Model {'metric': 'canberra', 'n_neighbors': 10, 'weights': 'uniform'}

The most important hyperparameter for KNN is the number of neighbors (n_neighbors).

Test values between at least 1 and 21, perhaps just the odd numbers.

n_neighbors in [1 to 21] It may also be interesting to test different distance metrics (metric) for choosing the composition of the neighborhood.

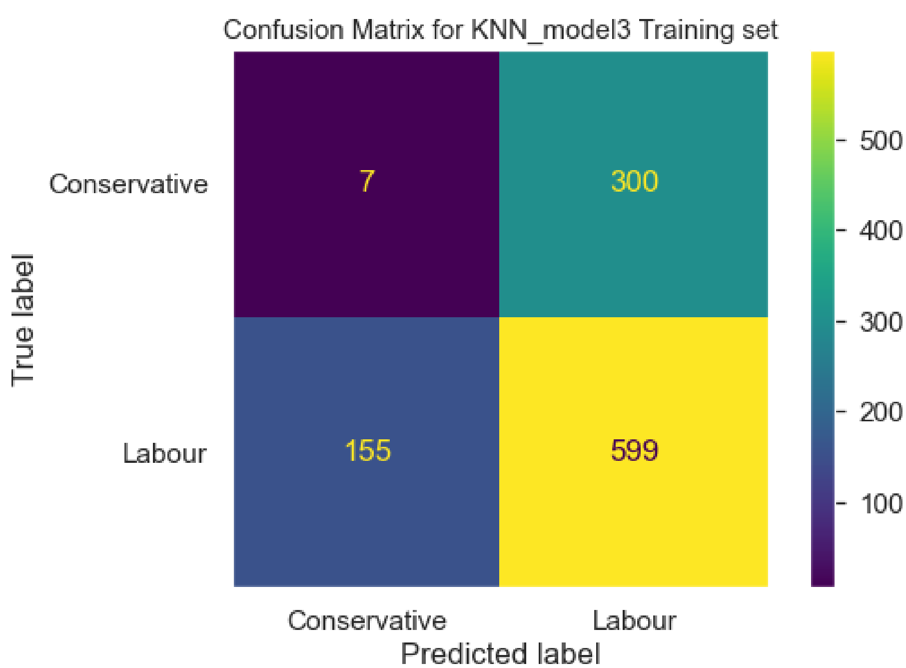
metric in [‘euclidean’, ‘manhattan’, ‘minkowski’] For a fuller list see:

sklearn.neighbors.DistanceMetric API It may also be interesting to test the contribution of members of the neighborhood via different weightings (weights).

weights in [‘uniform’, ‘distance’]

Train :

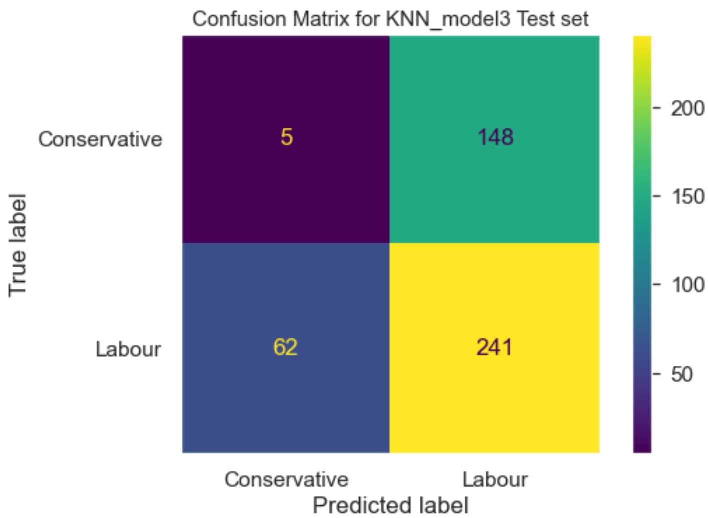
	precision	recall	f1-score	support
0	0.73	0.77	0.75	307
1	0.90	0.88	0.89	754
accuracy			0.85	1061
macro avg	0.82	0.83	0.82	1061
weighted avg	0.85	0.85	0.85	1061



Test :

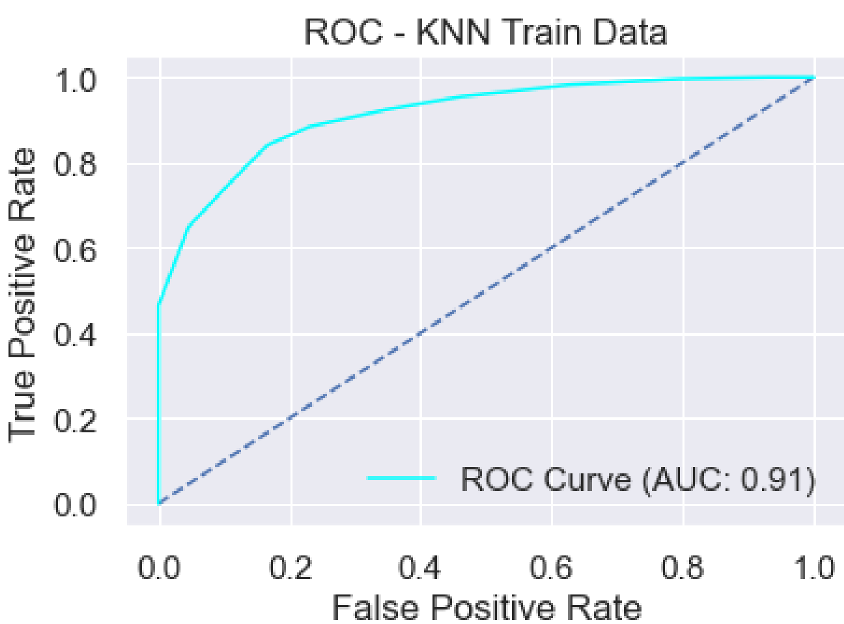
	precision	recall	f1-score	support
0	0.73	0.72	0.73	153
1	0.86	0.87	0.86	303
accuracy			0.82	456
macro avg	0.80	0.79	0.79	456

```
weighted avg       0.82      0.82      0.82       456
```



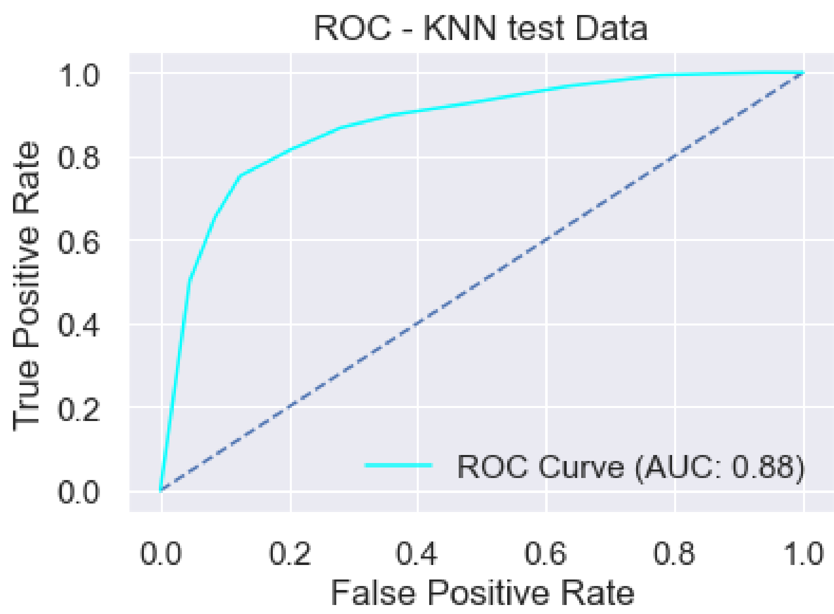
Train Roc :

```
KNN_train_auc 0.9148968800490761
```



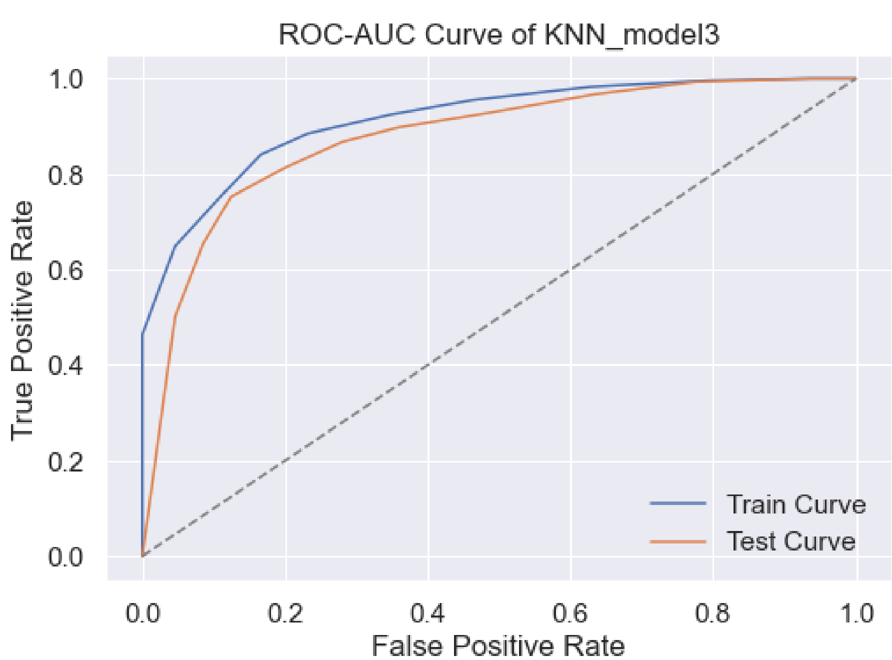
Test Roc :

```
KNN_test_auc 0.8767553225910827
```



Combined :

AUC for Training data = 0.9148968800490761
AUC for Test data = 0.8767553225910827



Naive Bayes

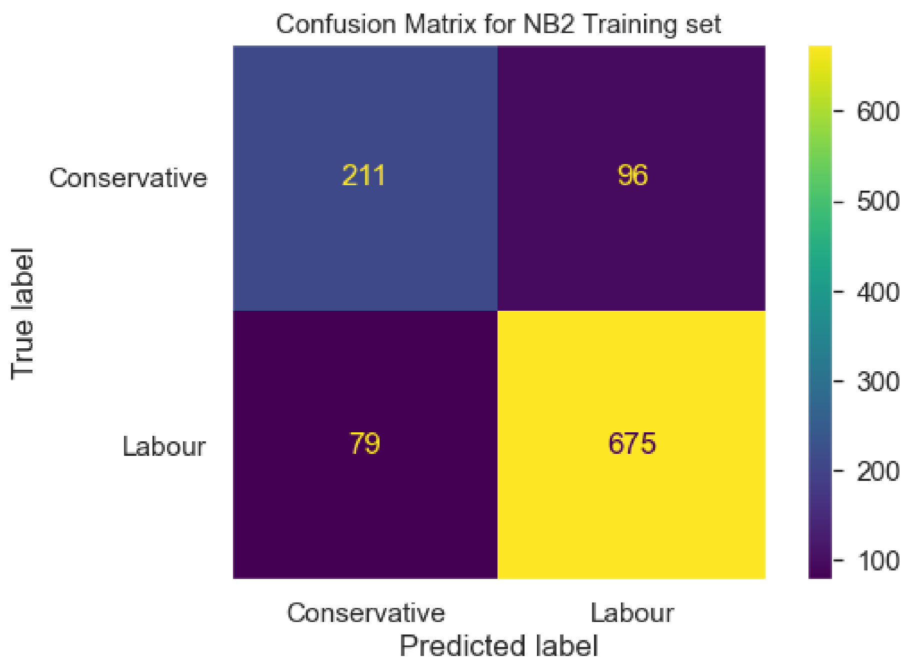
GaussianNB

```
GaussianNB(var_smoothing=0.0)
```

Var_smoothing (Variance smoothing) parameter specifies the portion of the largest variance of all features to be added to variances for stability of calculation.

Train:

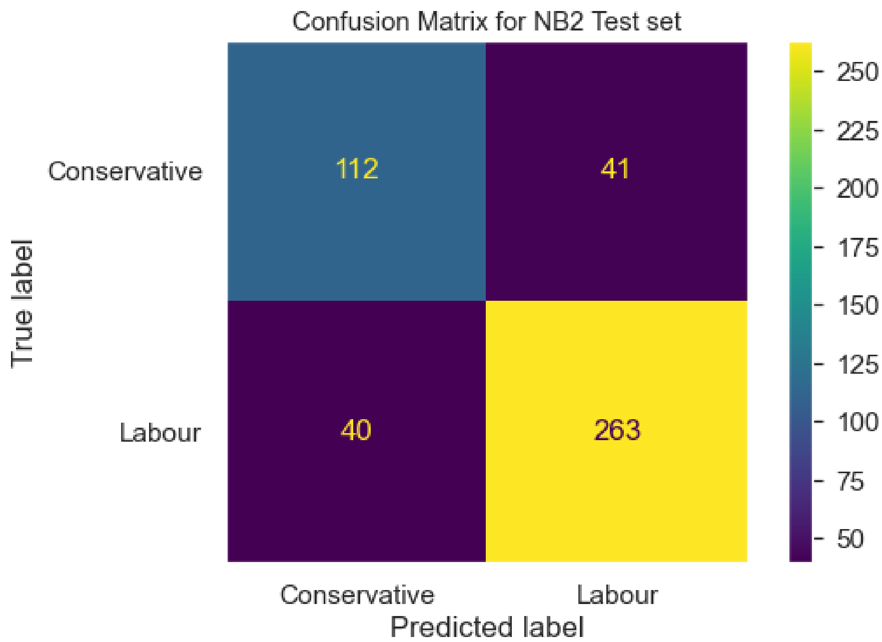
	precision	recall	f1-score	support
0	0.73	0.69	0.71	307
1	0.88	0.90	0.89	754
accuracy			0.84	1061
macro avg	0.80	0.79	0.80	1061
weighted avg	0.83	0.84	0.83	1061



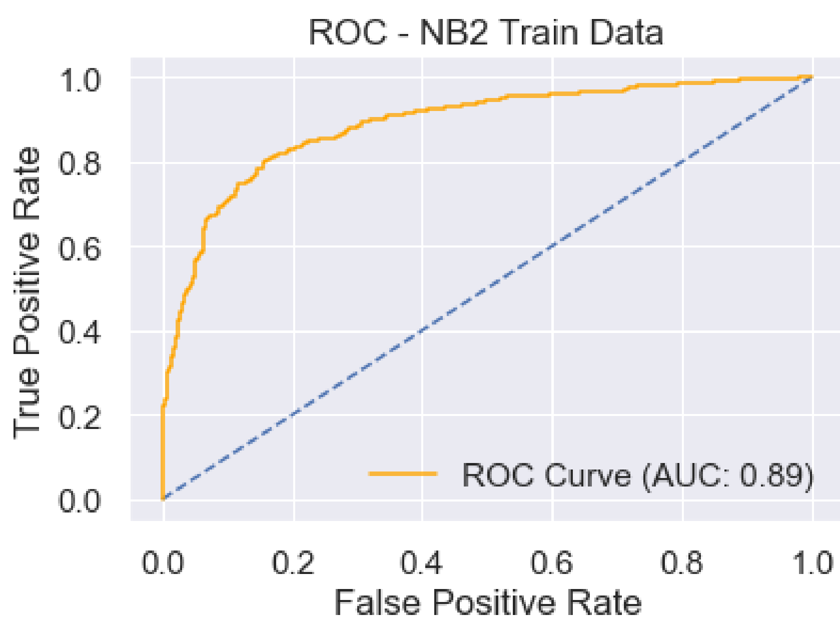
Test :

	precision	recall	f1-score	support
0	0.74	0.73	0.73	153
1	0.87	0.87	0.87	303
accuracy			0.82	456

```
   macro avg       0.80      0.80      0.80       456
weighted avg       0.82      0.82      0.82       456
```



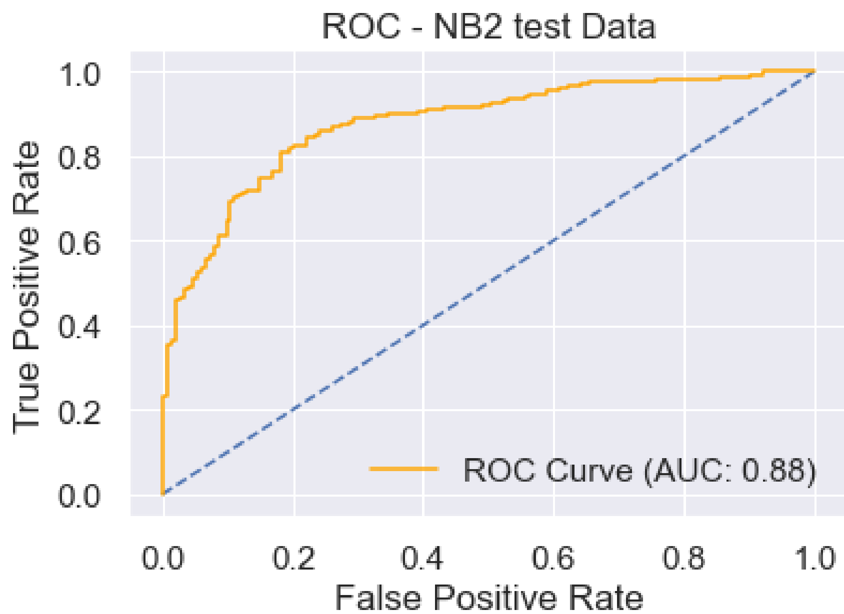
Train ROC:



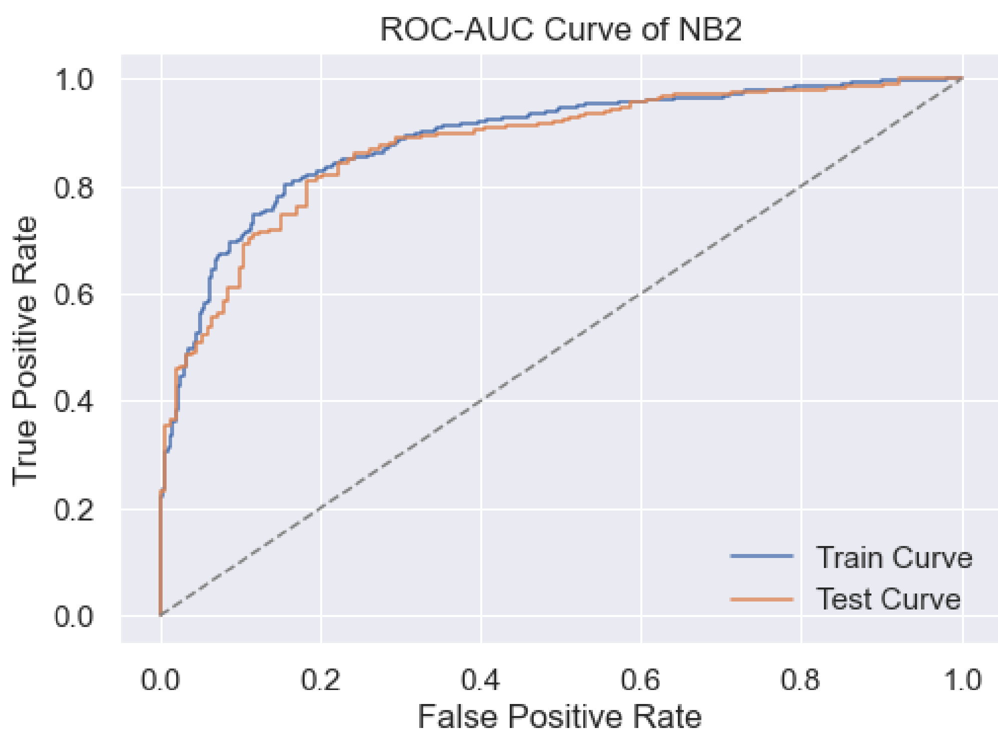

```
NB2_train_auc 0.8879375145802193
```

Test ROC:

```
NB2_test_auc 0.8763562630772882
```



Combined :




```
AUC for Training data = 0.8879375145802193
AUC for Test data = 0.8763562630772882
```

ADDITIONAL MODEL :

Support Vector Machine with GridsearchCV

GridSearchCV

```
GridSearchCV(cv=3, estimator=SVC(class_weight={0: 2.3, 1: 1}, probability=True),
             param_grid={'C': array([ 0.1       ,  0.1274275 ,  0.16237767,  0.20691
381,  0.26366509,
        0.33598183,  0.42813324,  0.54555948,  0.6951928 ,  0.88586679,
        1.12883789,  1.43844989,  1.83298071,  2.33572147,  2.97635144,
        3.79269019,  4.83293024,  6.15848211,  7.8475997 , 10.        ]),
                         'kernel': ['linear']})
```

estimator: SVC

```
SVC(class_weight={0: 2.3, 1: 1}, probability=True)
```

SVC

```
SVC(class_weight={0: 2.3, 1: 1}, probability=True)
```

Best Parameters from SVM Model {'C': 0.1, 'kernel': 'linear'}

The SVM algorithm, like gradient boosting, is very popular, very effective, and provides a large number of hyperparameters to tune.

Perhaps the first important parameter is the choice of kernel that will control the manner in which the input variables will be projected. There are many to choose from, but linear, polynomial, and RBF are the most common, perhaps just linear and RBF in practice.

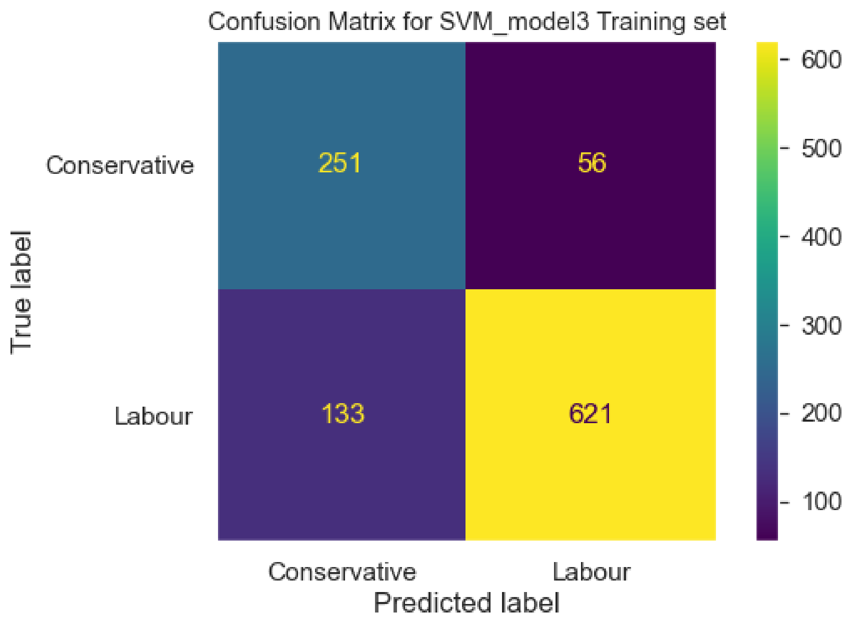
kernels in [‘linear’, ‘poly’, ‘rbf’, ‘sigmoid’] If the polynomial kernel works out, then it is a good idea to dive into the degree hyperparameter.

Another critical parameter is the penalty (C) that can take on a range of values and has a dramatic effect on the shape of the resulting regions for each class. A log scale might be a good starting point.

C in [100, 10, 1.0, 0.1, 0.001]

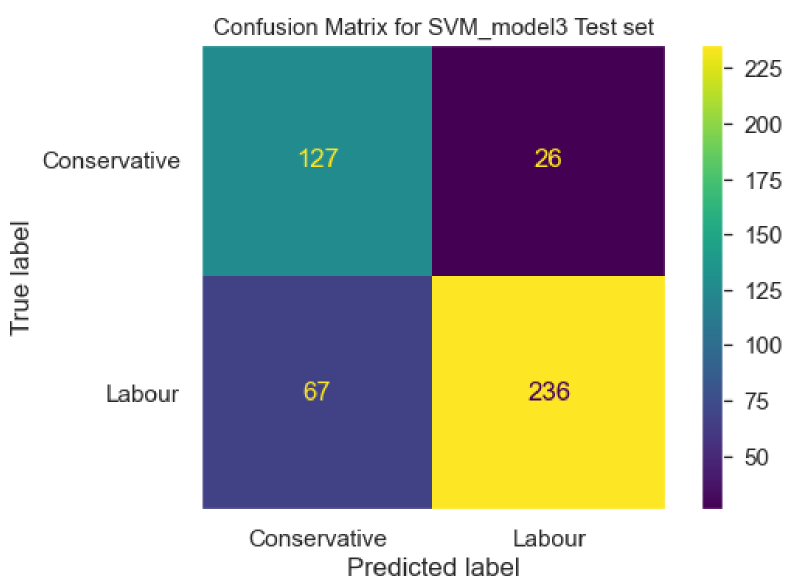
TRAIN :

```
              precision    recall  f1-score   support

           0       0.65      0.82      0.73       307
           1       0.92      0.82      0.87       754

    accuracy                           0.82      1061
   macro avg       0.79      0.82      0.80      1061
weighted avg       0.84      0.82      0.83      1061
```



TEST :

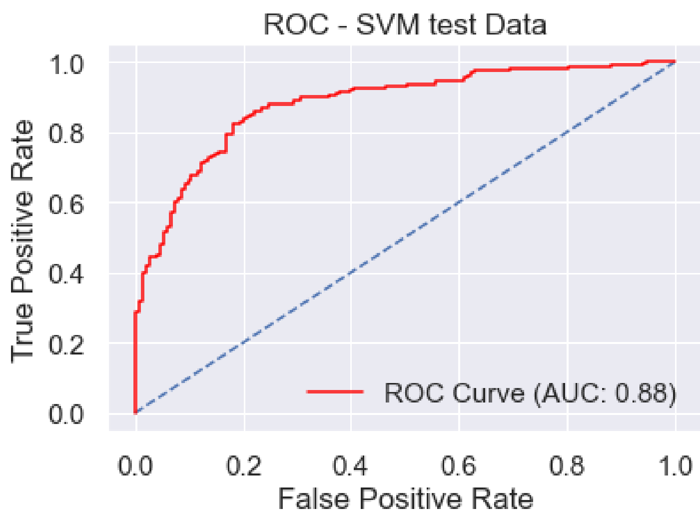
	precision	recall	f1-score	support
0	0.65	0.83	0.73	153
1	0.90	0.78	0.84	303
accuracy			0.80	456
macro avg	0.78	0.80	0.78	456
weighted avg	0.82	0.80	0.80	456



ROC - SVM Train Data

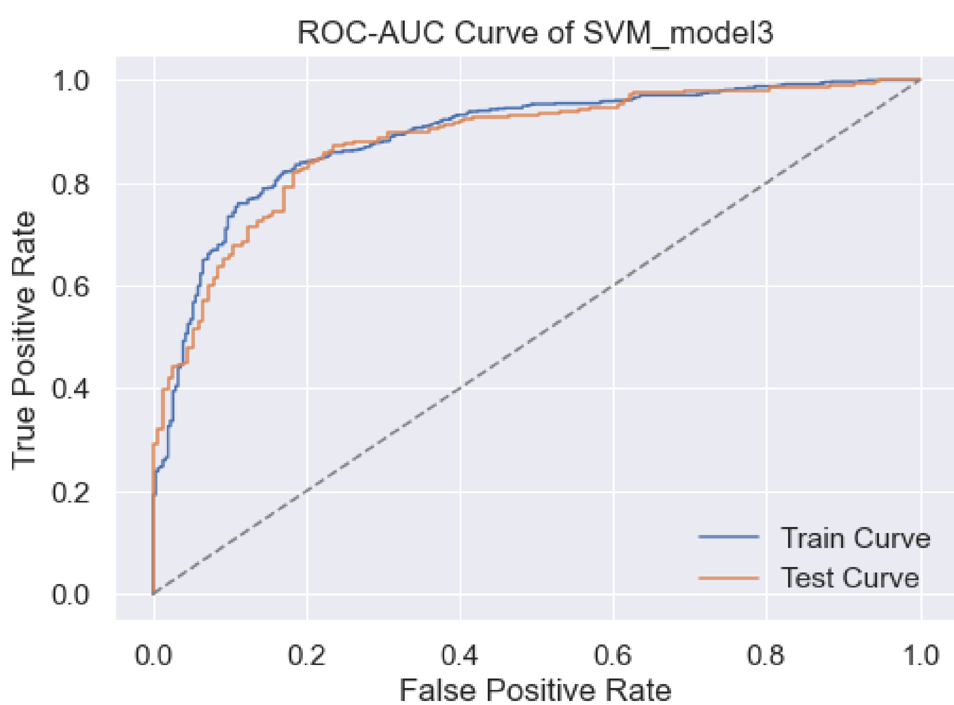

```
SVM_train_auc 0.8901277875219245
```

ROC - SVM test Data



SVM_test_auc 0.881037123320175

COMBINED :



AUC for Training data = 0.8901277875219245
AUC for Test data = 0.881037123320175

Bagging using RandomForest

BaggingClassifier

```
BaggingClassifier(base_estimator=RandomForestClassifier(class_weight={0: 4,
                                                                      1: 1.5},
                                                        min_samples_leaf=2,
                                                        min_samples_split=4),
                  n_estimators=50, random_state=1)
```

base_estimator: RandomForestClassifier

```
RandomForestClassifier(class_weight={0: 4, 1: 1.5}, min_samples_leaf=2,
                       min_samples_split=4)
```

RandomForestClassifier

```
RandomForestClassifier(class_weight={0: 4, 1: 1.5}, min_samples_leaf=2,
                       min_samples_split=4)
```

n_estimators The number of base estimators in the ensemble.

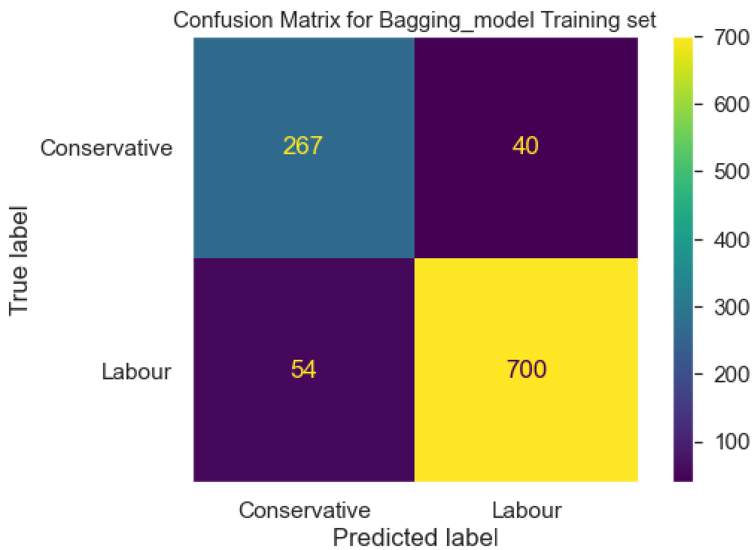
max_samples The number of samples to draw from X to train each base estimator (with replacement by default, see bootstrap for more details).

If int, then draw max_samples samples.

If float, then draw max_samples * X.shape[0] samples.

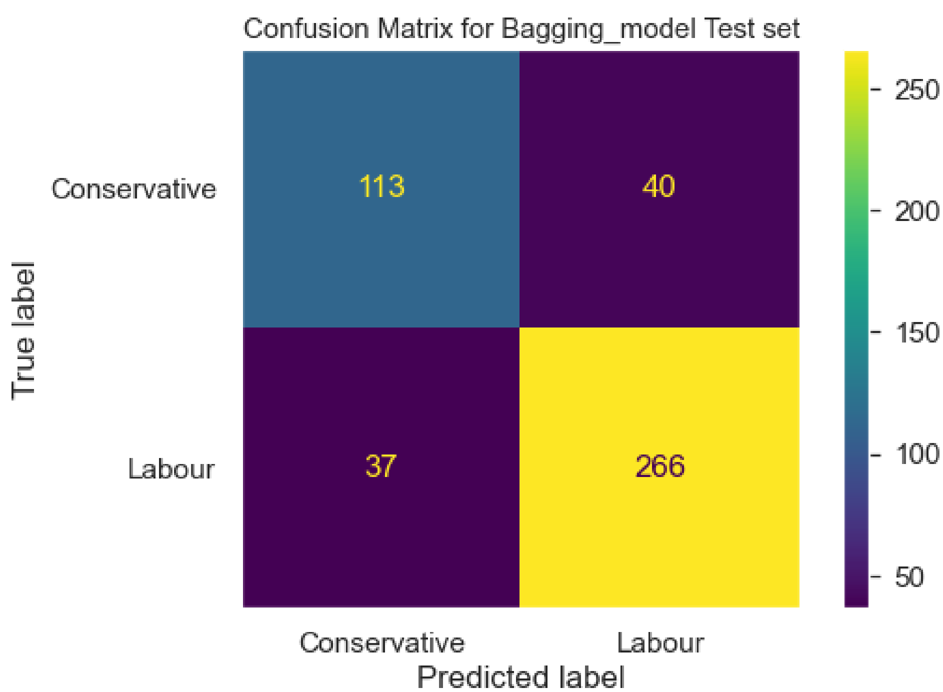
Train :

	precision	recall	f1-score	support
0	0.83	0.87	0.85	307
1	0.95	0.93	0.94	754
accuracy			0.91	1061
macro avg	0.89	0.90	0.89	1061
weighted avg	0.91	0.91	0.91	1061

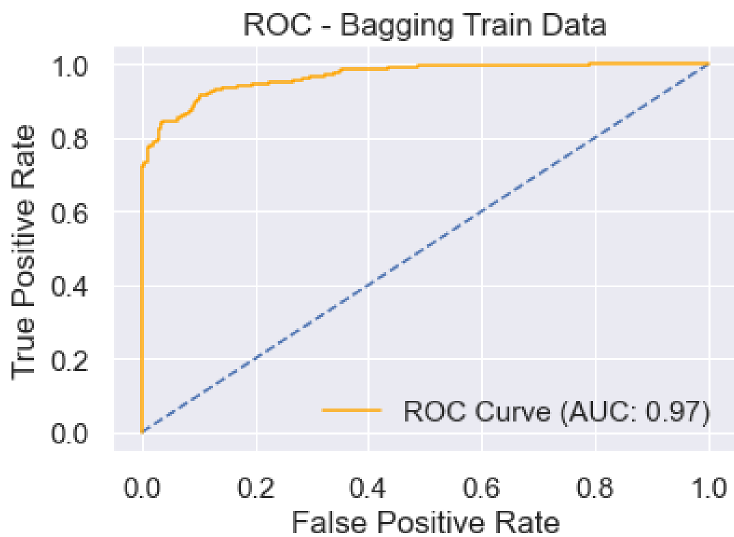


TEST :

	precision	recall	f1-score	support
0	0.75	0.74	0.75	153
1	0.87	0.88	0.87	303
accuracy			0.83	456
macro avg	0.81	0.81	0.81	456
weighted avg	0.83	0.83	0.83	456

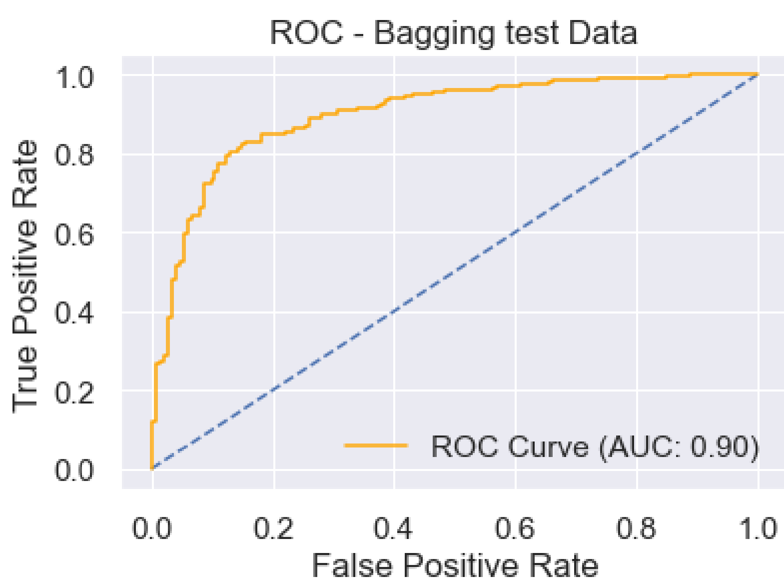


TRAIN BAGGING ROC :



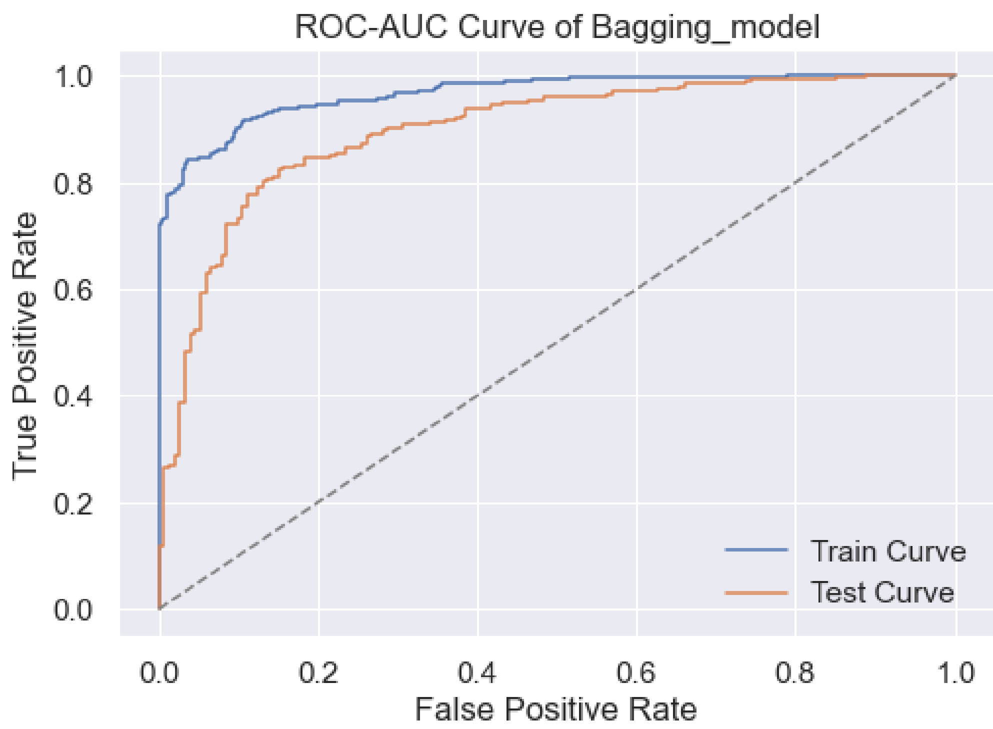
Bagging_train_auc 0.9678111958803861

TEST BAGGING ROC :



Bagging_test_auc 0.8981211846674864

COMBINED :




```
AUC for Training data = 0.9678111958803861
AUC for Test data = 0.8981211846674864
```

XGBOOST

XGBClassifier

```
XGBClassifier(base_score=0.5, booster='gbtree', callbacks=None,
              colsample_bylevel=1, colsample_bynode=1, colsample_bytree=1,
              early_stopping_rounds=None, enable_categorical=False,
              eval_metric=None, gamma=0, gpu_id=-1, grow_policy='depthwise',
              importance_type=None, interaction_constraints='',
              learning_rate=0.01, max_bin=256, max_cat_to_onehot=4,
              max_delta_step=0, max_depth=5, max_leaves=0, min_child_weight=3,
              missing=nan, monotone_constraints='()', n_estimators=1000,
              n_jobs=0, num_parallel_tree=1, predictor='auto', random_state=0,
              reg_alpha=0, reg_lambda=1, ...)
```

max_depth

Maximum depth of a tree. Increasing this value will make the model more complex and more likely to overfit. 0 indicates no limit on depth. Beware that XGBoost aggressively consumes memory when training a deep tree. exact tree method requires non-zero value.

range:

min_child_weight

Minimum sum of instance weight (hessian) needed in a child. If the tree partition step results in a leaf node with the sum of instance weight less than min_child_weight, then the building process will give up further partitioning. In linear regression task, this simply corresponds to minimum number of instances needed to be in each node. The larger min_child_weight is, the more conservative the algorithm will be.

The most important parameter for bagged decision trees is the number of trees (n_estimators).

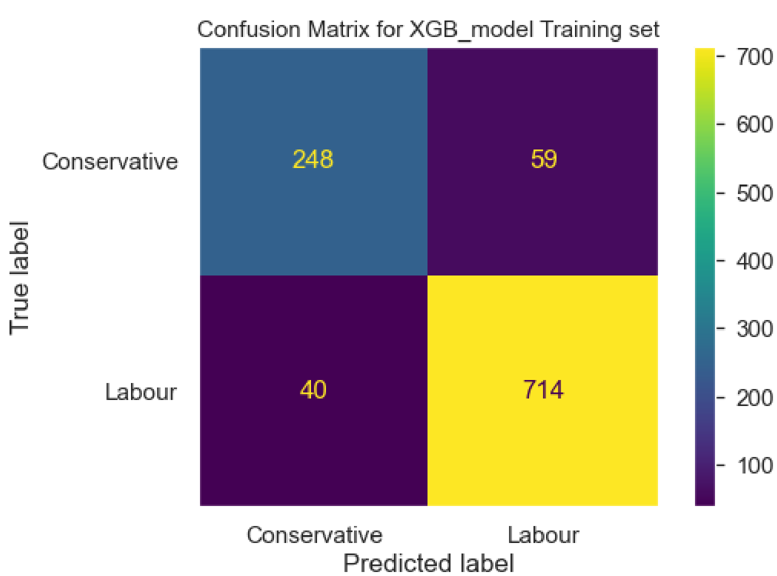
Ideally, this should be increased until no further improvement is seen in the model.

Good values might be a log scale from 10 to 1,000.

n_estimators in [10, 100, 1000]

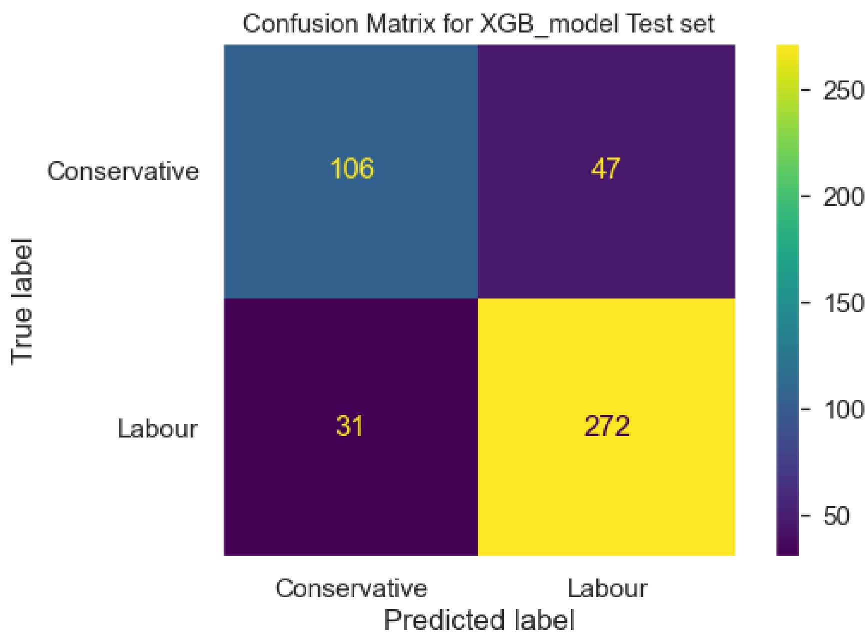
The learning_rate parameter can be set to control the weighting of new trees added to the model.

```
Train :

              precision    recall  f1-score   support

           0       0.86      0.81      0.83       307
           1       0.92      0.95      0.94       754

    accuracy                           0.91      1061
   macro avg       0.89      0.88      0.88      1061
weighted avg       0.91      0.91      0.91      1061
```

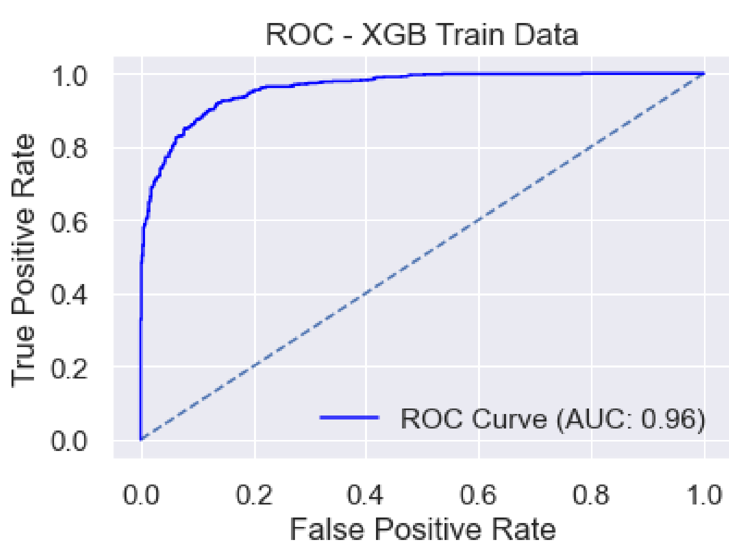


Test :

	precision	recall	f1-score	support
0	0.77	0.69	0.73	153
1	0.85	0.90	0.87	303
accuracy			0.83	456
macro avg	0.81	0.80	0.80	456
weighted avg	0.83	0.83	0.83	456



ROC - XGB Train Data

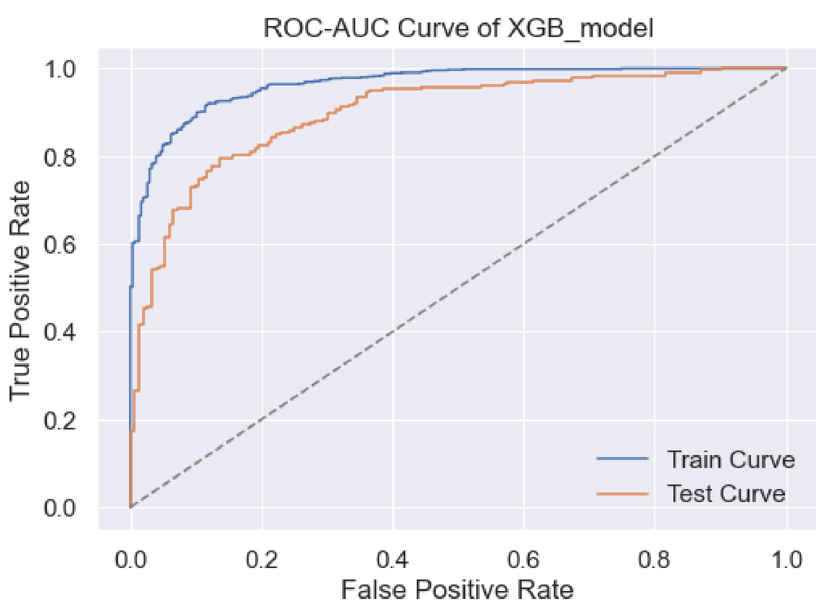


```
XGB_train_auc 0.9599633079807781
```

ROC - XGB test Data

`XGB_test_auc 0.8986604542807222`

`Combined:`




```
AUC for Training data = 0.9647072291967271
AUC for Test data = 0.8986604542807222
```

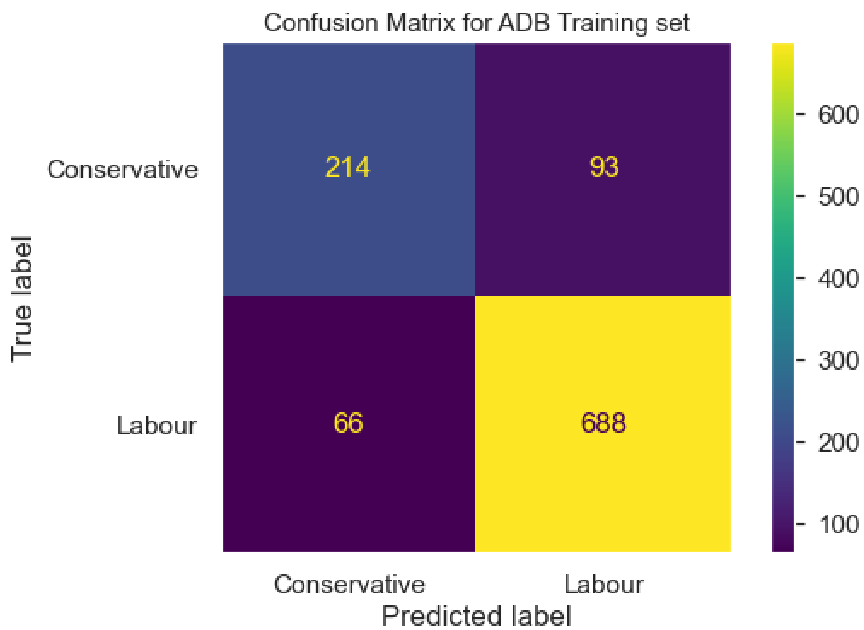
Ada Boost

```
AdaBoostClassifier(n_estimators=100, random_state=1)
```

The maximum number of estimators at which boosting is terminated. In case of perfect fit, the learning procedure is stopped early.

Train :

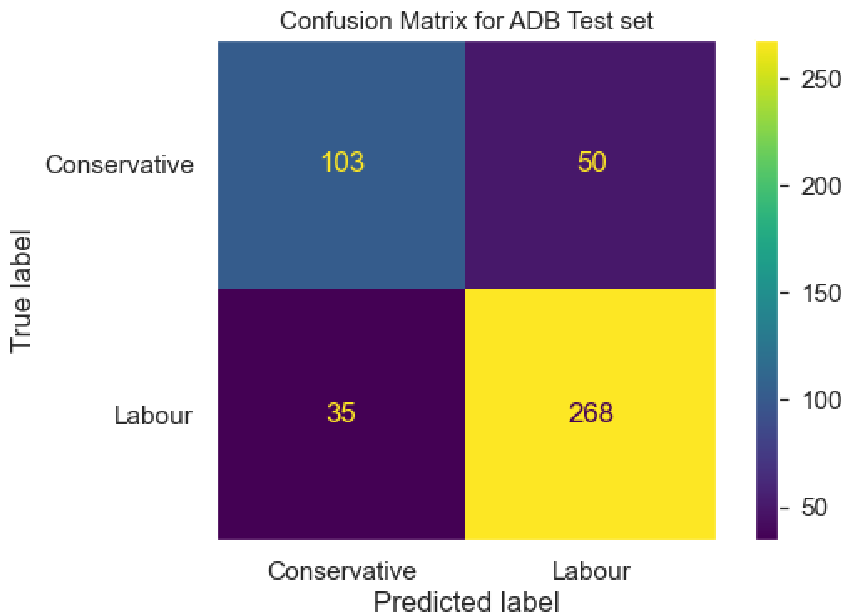
	precision	recall	f1-score	support
0	0.86	0.81	0.83	307
1	0.92	0.95	0.94	754
accuracy			0.91	1061
macro avg	0.89	0.88	0.88	1061
weighted avg	0.91	0.91	0.91	1061



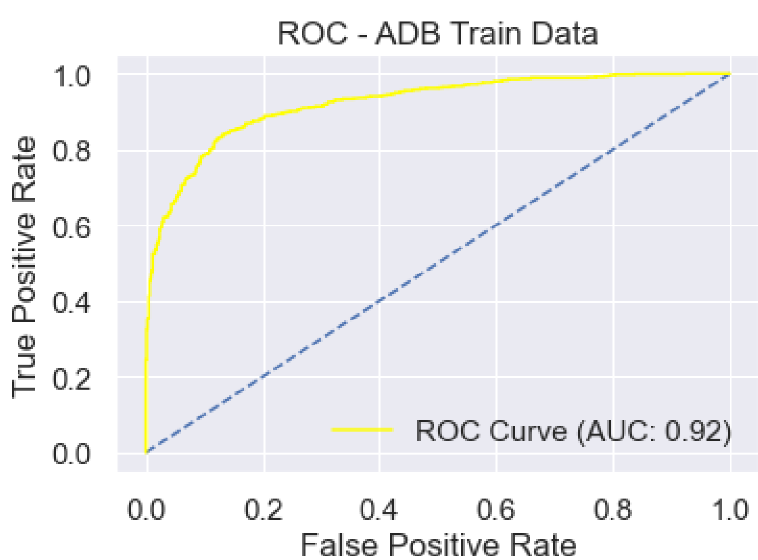
Test :

	precision	recall	f1-score	support
0	0.77	0.69	0.73	153
1	0.85	0.90	0.87	303
accuracy			0.83	456

```
   macro avg       0.81      0.80      0.80       456
weighted avg       0.83      0.83      0.83       456
```

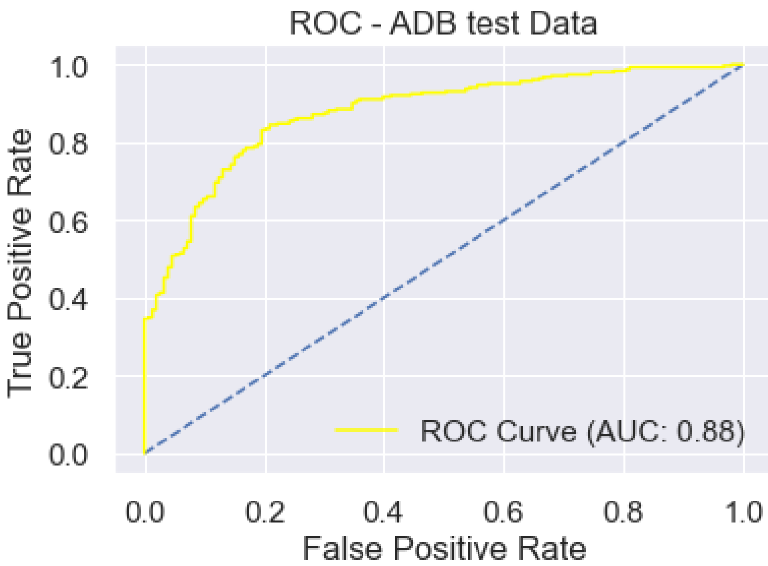


TRAIN ROC



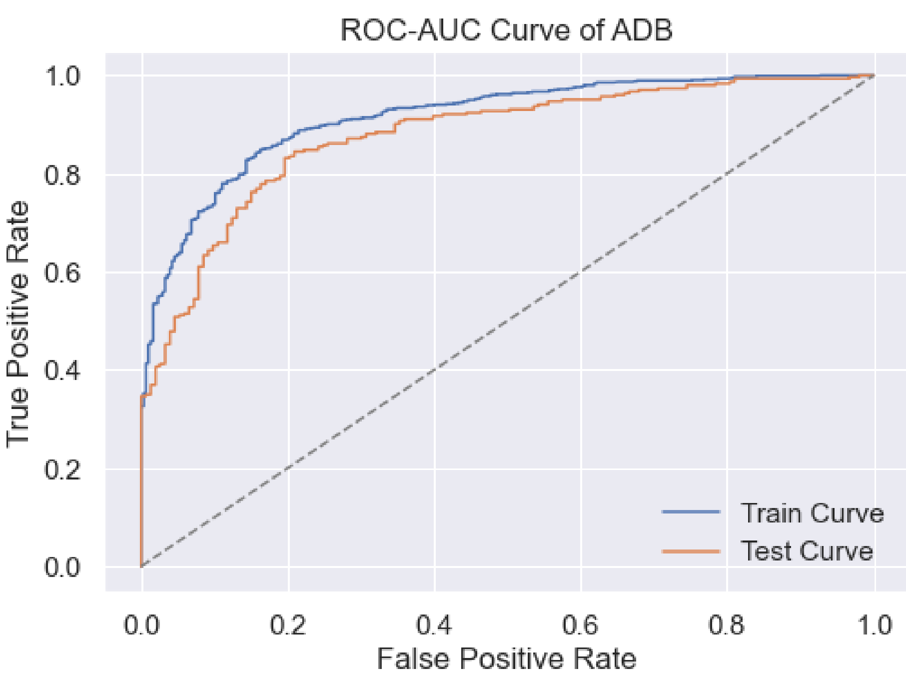

```
ADB_train_auc 0.9214701081411958
```

TEST ROC :



ADB_test_auc 0.8773808753424363

COMBINED :



AUC for Training data = 0.9148061586846267
AUC for Test data = 0.8773808753424363

Gradient Boosting Classifier

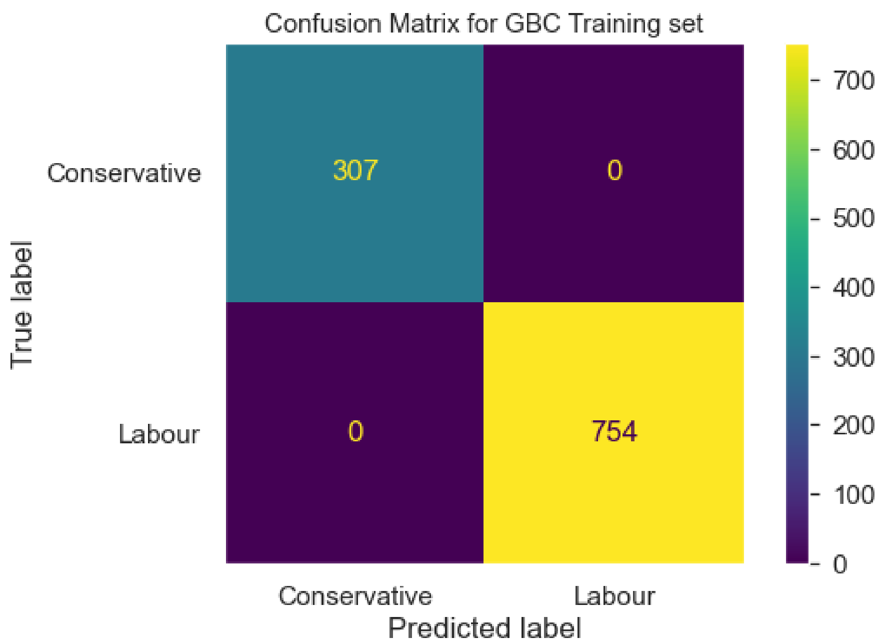
```
GradientBoostingClassifier(max_depth=10, n_estimators=500)
```

The number of trees in the model (n_estimators)

The depth of each tree (max_depth)

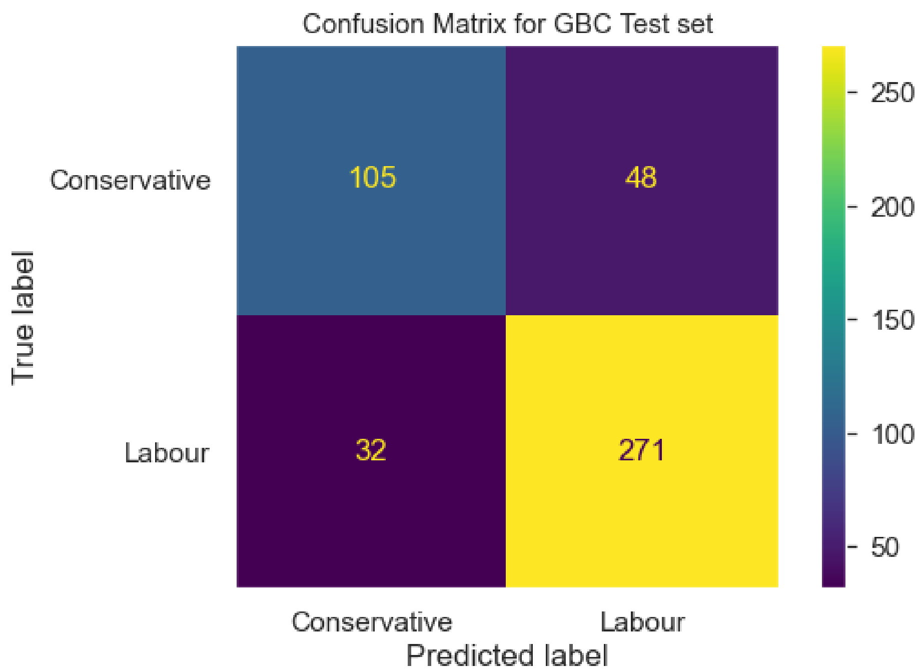
TRAIN :

	precision	recall	f1-score	support
0	0.86	0.81	0.83	307
1	0.92	0.95	0.94	754
accuracy			0.91	1061
macro avg	0.89	0.88	0.88	1061
weighted avg	0.91	0.91	0.91	1061

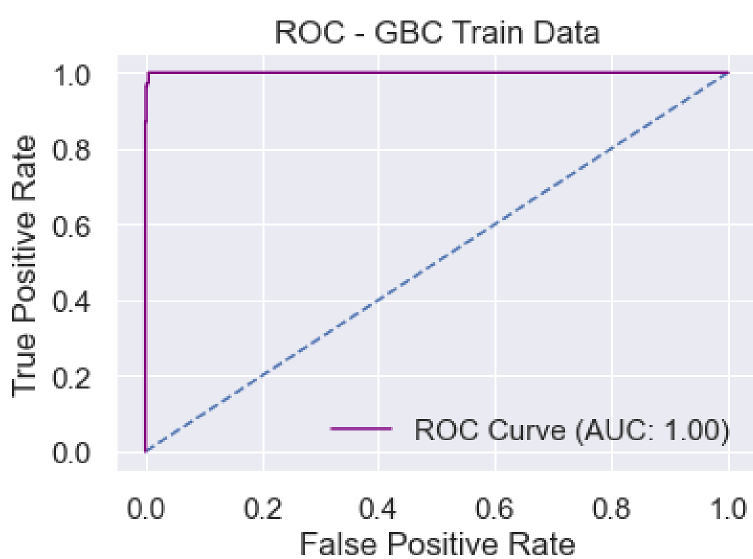


TEST :

	precision	recall	f1-score	support
0	0.77	0.69	0.73	153
1	0.85	0.90	0.87	303
accuracy			0.83	456
macro avg	0.81	0.80	0.80	456
weighted avg	0.83	0.83	0.83	456

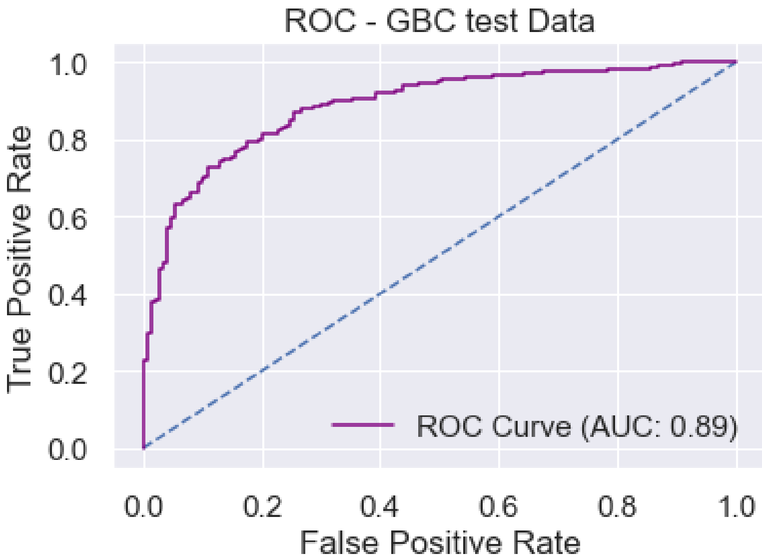


ROC - GBC train Data



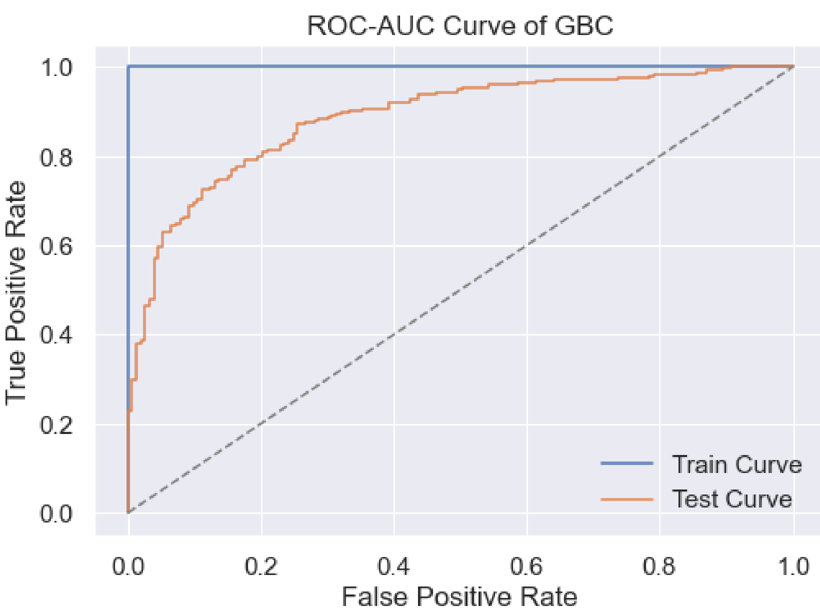

```
GBC_train_auc 0.9997097707012643
```

ROC - GBC test Data




```
GBC_test_auc 0.8865376733751806
```

Combined:




```
AUC for Training data = 1.0
AUC for Test data = 0.8773808753424363
```

1.7 Performance Metrics: Check the performance of Predictions on Train and Test sets using Accuracy, Confusion Matrix, Plot ROC curve and get ROC_AUC score for each model, classification report (4 pts) Final Model

TRAIN :

	Logit Train	LDA Train	KNN Train	NB2 Train	SVM Train	Bagging Train	XGB Train	ADB Train	GBC Train
Accuracy	0.83	0.83	0.85	0.84	0.82	0.91	0.91	0.85	1.0
AUC	0.89	0.89	0.91	0.89	0.89	0.97	0.96	0.92	1.0
Recall-0	0.79	0.65	0.77	0.69	0.82	0.87	0.81	0.81	1.0
Recall-1	0.85	0.91	0.88	0.90	0.82	0.93	0.95	0.95	1.0
Precision-0	0.68	0.74	0.73	0.73	0.65	0.83	0.86	0.86	1.0
Precision-1	0.91	0.86	0.90	0.88	0.92	0.95	0.92	0.92	1.0
F1 Score-0	0.73	0.69	0.75	0.71	0.73	0.85	0.83	0.83	1.0
F1 Score-1	0.88	0.89	0.89	0.89	0.87	0.94	0.94	0.94	1.0

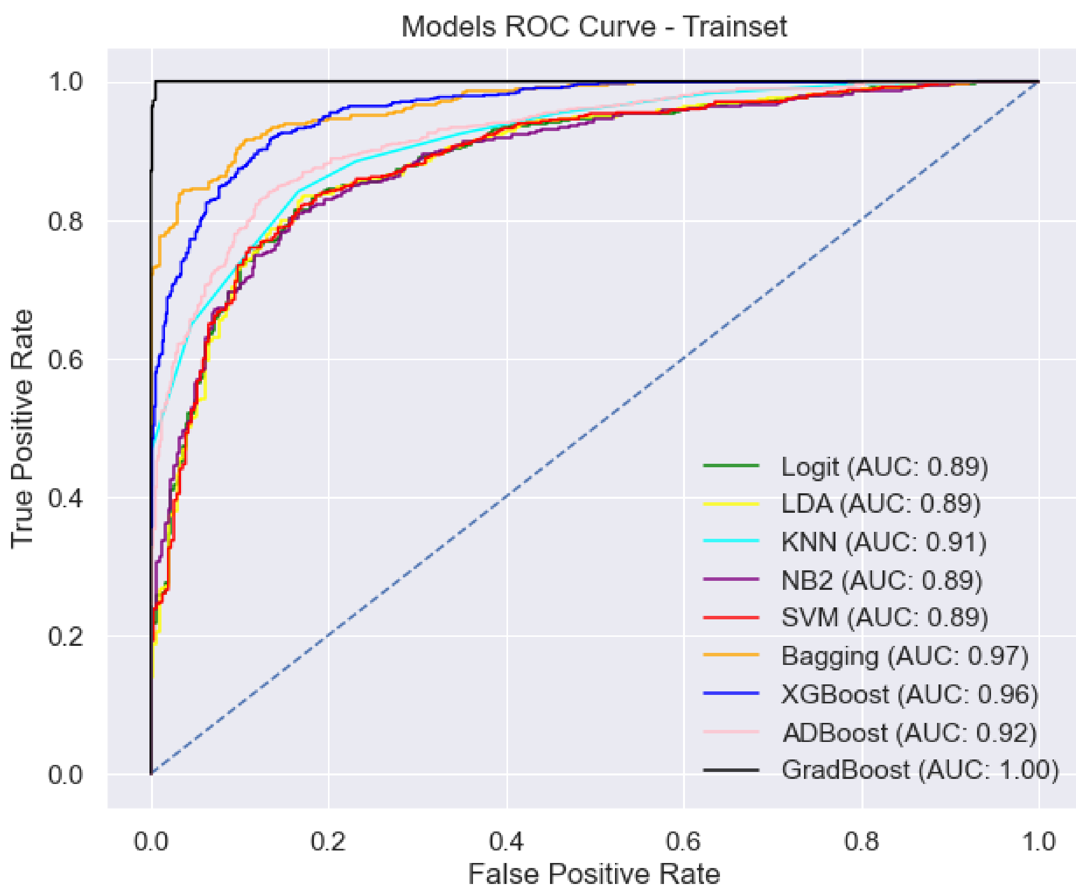
TEST :

	Logit Test	LDA Test	KNN Test	NB2 Test	SVM Test	Bagging Test	XGB Test	ADB Test	GBC Test
Accuracy	0.82	0.83	0.82	0.82	0.80	0.83	0.83	0.81	0.82
AUC	0.88	0.89	0.88	0.88	0.88	0.90	0.90	0.88	0.89
Recall-0	0.81	0.73	0.72	0.73	0.83	0.74	0.69	0.69	0.69

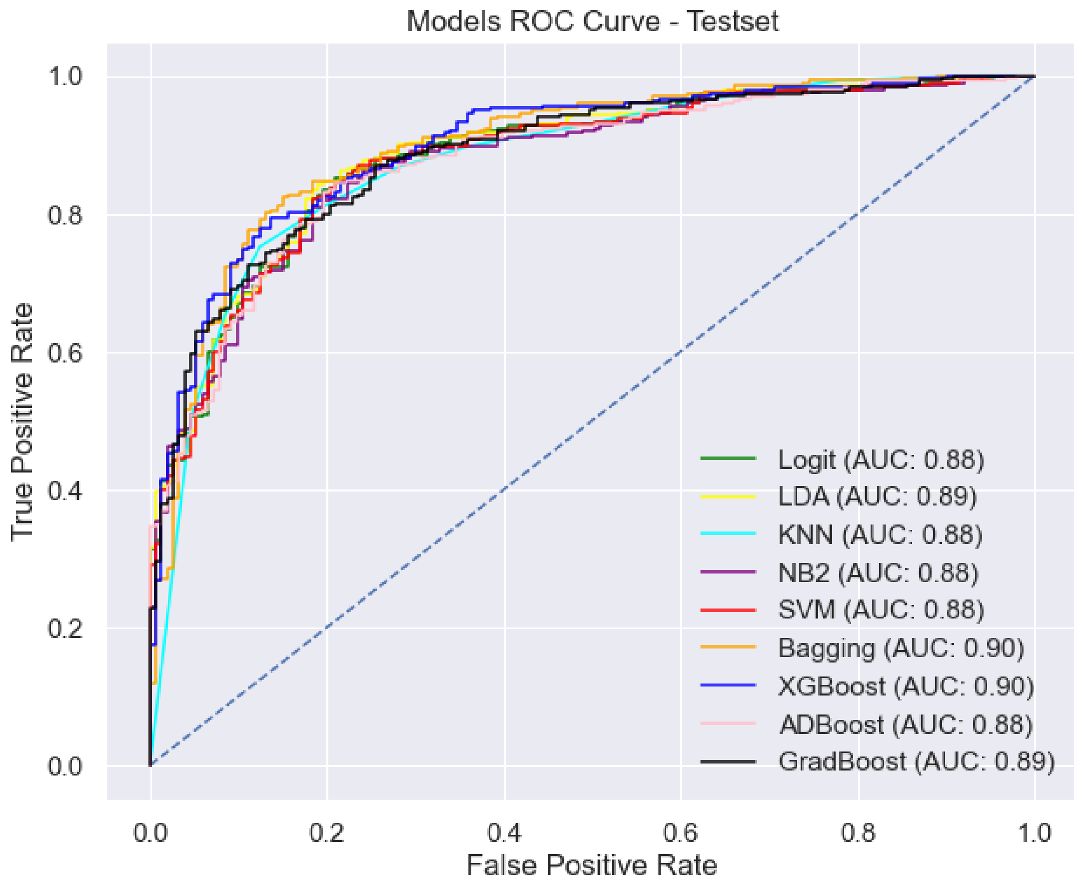
	Logit Test	LDA Test	KNN Test	NB2 Test	SVM Test	Bagging Test	XGB Test	ADB Test	GBC Test
Recall-1	0.82	0.89	0.87	0.87	0.78	0.88	0.90	0.90	0.89
Precision-0	0.70	0.77	0.73	0.74	0.65	0.75	0.77	0.77	0.77
Precision-1	0.90	0.86	0.86	0.87	0.90	0.87	0.85	0.85	0.85
F1 Score-0	0.75	0.74	0.73	0.73	0.73	0.75	0.73	0.73	0.72
F1 Score-1	0.86	0.88	0.86	0.87	0.84	0.87	0.87	0.87	0.87

ROC AUC OF EVERY MODEL :

Train:



TEST :



In terms of model selection i will be choosing bagging as my go to model because of the accuracy value and the vale of recall these both value have a blanced kind of thing between them also the accuracy is the most high in this model as we need accuray in our selection process so we can know the definitive answer. as bagging model recall score helps us to know that what amount of votes are really going to the supposed party that we are finding about XGB model is also good but in comparison of both accuracy and recall score XGB model is performing quite less in camparision of bagging model. As we can saw that by the recall score of the bagging model that most of the prediction done by bagging is reliable then the others. So thats why as our bagging model is giving us most amount of correct predictions and by studying the models and confusion matrix most amount of vote or voters are favouring tha labour class or party. So, by this we have properly predicted that which party is going to win as most of the other models are also favourig the labour class only so you can also get a idea of the winning party.

1.8 Based on these predictions, what are the insights?

Here, by studying the model we came to know that most the population is opting for labour class. So, we can say that labour class is going to win on the basis of the prediction done by our models, as most of the voters or votes are from labour class only, as we not have enought information about the work of labour class and conservation class i may not be able to tell the suggestion of how the respected class must improve. From my prediction, labour class is almost covering 70-85% of the seats. Also the proportion of voters or votes from conservative party is not much, that is one reason that they arent able to win or acquire seats for themselves my suggest would to add the numbers to their party. we can see that labour party have overwhelming strenght in terms of votes or voters that is the reason they are going to win as per prediction done by our model because most of the votes are going to the labour class.

Problem 2:

In this particular project, we are going to work on the inaugural corpora from the nltk in Python. We will be looking at the following speeches of the Presidents of the United States of America: President Franklin D. Roosevelt in 1941 President John F. Kennedy in 1961 President Richard Nixon in 1973

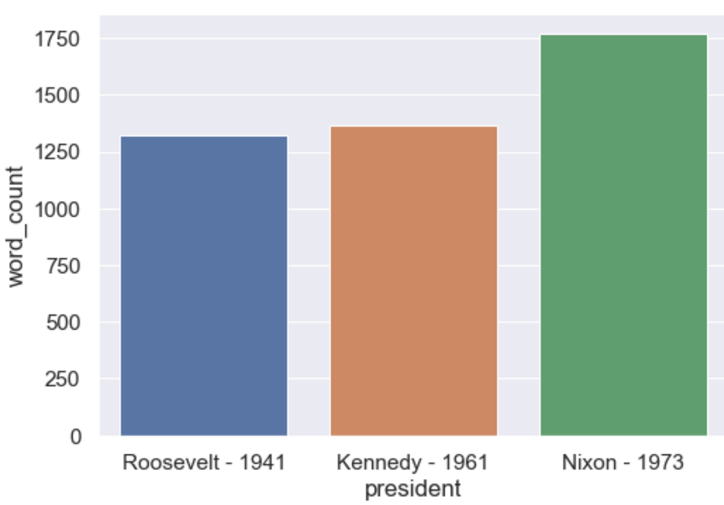
TABLE :

	president	text
1941-Roosevelt	Roosevelt - 1941	On each national day of inauguration since 178...
1961-Kennedy	Kennedy - 1961	Vice President Johnson, Mr. Speaker, Mr. Chief...
1973-Nixon	Nixon - 1973	Mr. Vice President, Mr. Speaker, Mr. Chief Jus...

2.1 Find the number of characters, words, and sentences for the mentioned documents.

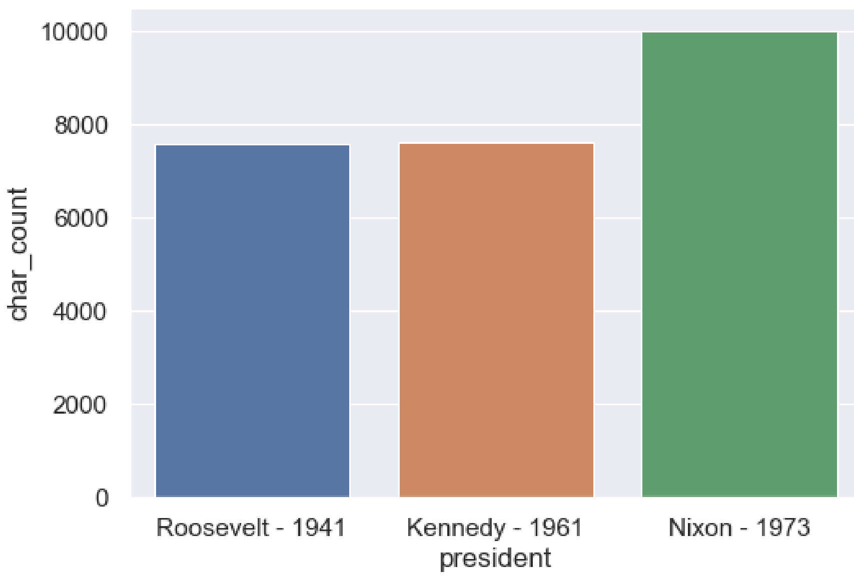
Number of words

	president	text	word_count
1941-Roosevelt	Roosevelt - 1941	On each national day of inauguration since 178...	1323
1961-Kennedy	Kennedy - 1961	Vice President Johnson, Mr. Speaker, Mr. Chief...	1364
1973-Nixon	Nixon - 1973	Mr. Vice President, Mr. Speaker, Mr. Chief Jus...	1769



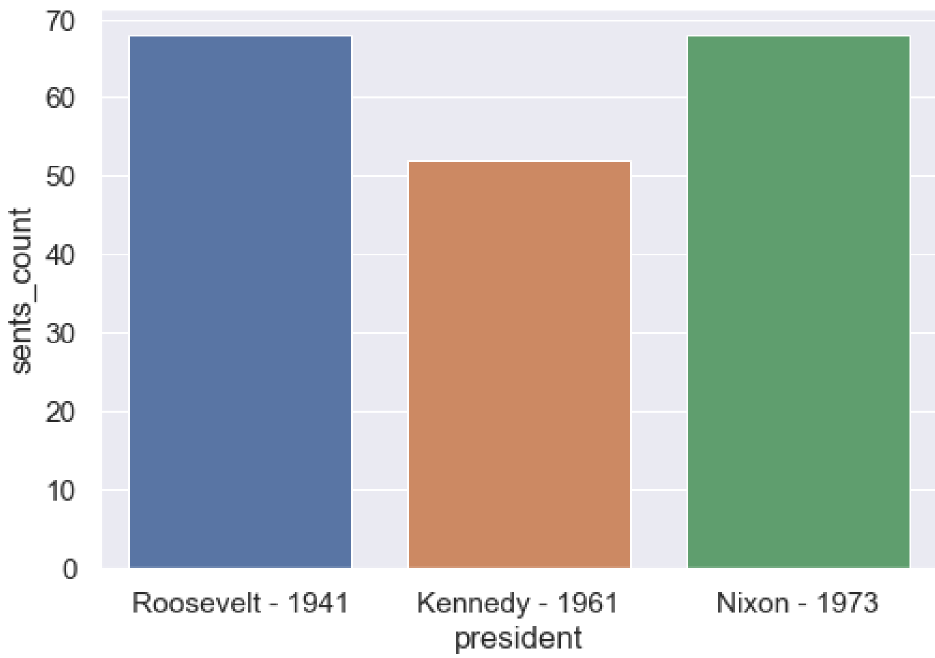
Number of characters

	president	text	word_count	char_count
1941-Roosevelt	Roosevelt - 1941	On each national day of inauguration since 178...	1323	7571
1961-Kennedy	Kennedy - 1961	Vice President Johnson, Mr. Speaker, Mr. Chief...	1364	7618
1973-Nixon	Nixon - 1973	Mr. Vice President, Mr. Speaker, Mr. Chief Jus...	1769	9991



Number of sentences

	president	text	word_count	char_count	sents_count
1941-Roosevelt	Roosevelt - 1941	On each national day of inauguration since 178...	1323	7571	68
1961-Kennedy	Kennedy - 1961	Vice President Johnson, Mr. Speaker, Mr. Chief...	1364	7618	52
1973-Nixon	Nixon - 1973	Mr. Vice President, Mr. Speaker, Mr. Chief Jus...	1769	9991	68



2.2 Remove all the stopwords from all three speeches

Lower case conversion

```
1941-Roosevelt    on each national day of inauguration since 178
...
1961-Kennedy      vice president johnson, mr. speaker, mr. chief
...
1973-Nixon        mr. vice president, mr. speaker, mr. chief jus
...
```

Remove punctuation

```
1941-Roosevelt    on each national day of inauguration since 178
...
1961-Kennedy      vice president johnson mr speaker mr chief jus
...
1973-Nixon        mr vice president mr speaker mr chief justice
...
```

Removing Stopwords :

	president	text	word_count	char_count	sents_count
1941-Roosevelt	Roosevelt - 1941	national day inauguration since 1789 people re...	1323	7571	68
1961-Kennedy	Kennedy - 1961	vice president johnson speaker chief justice p...	1364	7618	52
1973-Nixon	Nixon - 1973	vice president speaker chief justice senator c...	1769	9991	68

1. Speech of president Roosevelt without stopwords

```
['national day inauguration since 1789 people renewed sense dedication united states wash
ingtons day task people create weld together nation lincolns day task people preserve nat
ion disruption within day task people save nation institutions disruption without come ti
me midst swift happenings pause moment take stock recall place history rediscover may ris
k real peril inaction lives nations determined count years lifetime human spirit life man
threescore years ten little little less life nation fullness measure live men doubt men b
elieve democracy form government frame life limited measured kind mystical artificial fat
e unexplained reason tyranny slavery become surging wave future freedom ebbing tide ameri
cans know true eight years ago life republic seemed frozen fatalistic terror proved true 
midst shock acted acted quickly boldly decisively later years living years fruitful years
people democracy brought greater security hope better understanding lifes ideals measured
material things vital present future experience democracy successfully survived crisis ho
me put away many evil things built new structures enduring lines maintained fact democrac
y action taken within threeway framework constitution united states coordinate branches g
overnment continue freely function bill rights remains inviolate freedom elections wholly
maintained prophets downfall american democracy seen dire predictions come naught democra
cy dying know seen reviveand grow know cannot die built unhampered initiative individual 
men women joined together common enterprise enterprise undertaken carried free expression
free majority know democracy alone forms government enlists full force mens enlightened k
now democracy alone constructed unlimited civilization capable infinite progress improvem
ent human life know look surface sense still spreading every continent humane advanced en
d unconquerable forms human society nation like person bodya body must fed clothed housed
invigorated rested manner measures objectives time nation like person mind mind must kept
informed alert must know understands hopes needs neighbors nations live within narrowing 
circle world nation like person something deeper something permanent something larger sum
parts something matters future calls forth sacred guarding present thing find difficult e
ven impossible hit upon single simple word yet understand spirit faith america product ce
nturies born multitudes came many lands high degree mostly plain people sought early late
find freedom freely democratic aspiration mere recent phase human history human history p
ermeated ancient life early peoples blazed anew middle ages written magna charta americas
```

impact irresistible america new world tongues peoples continent newfound land came believ
ed could create upon continent new life life new freedom vitality written mayflower compa
ct declaration independence constitution united states gettysburg address first came carr
y longings spirit millions followed stock sprang moved forward constantly consistently to
ward ideal gained stature clarity generation hopes republic cannot forever tolerate eithe
r undeserved poverty selfserving wealth know still far go must greatly build security opp
ortunity knowledge every citizen measure justified resources capacity land enough achieve
purposes alone enough clothe feed body nation instruct inform mind also spirit three grea
test spirit without body mind men know nation could live spirit america killed even thoug
h nations body mind constricted alien world lived america know would perished spirit fait
h speaks daily lives ways often unnoticed seem obvious speaks capital nation speaks proce
sses governing sovereignties 48 states speaks counties cities towns villages speaks natio
ns hemisphere across seas enslaved well free sometimes fail hear heed voices freedom priv
ilege freedom old old story destiny america proclaimed words prophecy spoken first presid
ent first inaugural 1789 words almost directed would seem year 1941 preservation sacred f
ire liberty destiny republican model government justly considered deeply finally staked e
xperiment intrusted hands american people lose sacred fireif smothered doubt fear reject
destiny washington strove valiantly triumphantly establish preservation spirit faith nati
on furnish highest justification every sacrifice may make cause national defense face gre
at perils never encountered strong purpose protect perpetuate integrity democracy muster
spirit america faith america retreat content stand still americans go forward service cou
ntry god']

2. Speech of president Kennedy without stopwords

['vice president johnson speaker chief justice president eisenhower vice president nixon
president truman reverend clergy fellow citizens observe today victory party celebration
freedom symbolizing end well beginning signifying renewal well change sworn almighty god
solemn oath forebears l prescribed nearly century three quarters ago world different man
holds mortal hands power abolish forms human poverty forms human life yet revolutionary b
eliefs forebears fought still issue around globe belief rights man come generosity state
hand god dare forget today heirs first revolution word go forth time place friend foe ali
ke torch passed new generation americans born century tempered war disciplined hard bitte
r peace proud ancient heritage unwilling witness permit slow undoing human rights nation
always committed committed today home around world every nation know whether wishes well
ill pay price bear burden meet hardship support friend oppose foe order assure survival s
uccess liberty much pledge old allies whose cultural spiritual origins share pledge loyal
ty faithful friends united little cannot host cooperative ventures divided little dare me
et powerful challenge odds split asunder new states welcome ranks free pledge word one fo
rm colonial control passed away merely replaced far iron tyranny always expect find suppo
rting view always hope find strongly supporting freedom remember past foolishly sought po
wer riding back tiger ended inside peoples huts villages across globe struggling break bo
nds mass misery pledge best efforts help help whatever period required communists may see
k votes right free society cannot help many poor cannot save rich sister republics south
border offer special pledge convert good words good deeds new alliance progress assist fr
ee men free governments casting chains poverty peaceful revolution hope cannot become pre
y hostile powers neighbors know join oppose aggression subversion anywhere americas every

power know hemisphere intends remain master house world assembly sovereign states united nations last best hope age instruments war far outpaced instruments peace renew pledge su pportto prevent becoming merely forum invective strengthen shield new weak enlarge area w rit may run finally nations would make adversary offer pledge request sides begin anew qu est peace dark powers destruction unleashed science engulf humanity planned accidental se lfdestruction dare tempt weakness arms sufficient beyond doubt certain beyond doubt never employed neither two great powerful groups nations take comfort present course sides over burdened cost modern weapons rightly alarmed steady spread deadly atom yet racing alter u ncertain balance terror stays hand mankinds final war begin anew remembering sides civili ty sign weakness sincerity always subject proof never negotiate fear never fear negotiate sides explore problems unite instead belaboring problems divide sides first time formulat e serious precise proposals inspection control arms bring absolute power destroy nations absolute control nations sides seek invoke wonders science instead terrors together explo re stars conquer deserts eradicate disease tap ocean depths encourage arts commerce sides unite heed corners earth command isaiah undo heavy burdens oppressed go free beachhead co operation may push back jungle suspicion sides join creating new endeavor new balance pow er new world law strong weak secure peace preserved finished first 100 days finished firs t 1000 days life administration even perhaps lifetime planet begin hands fellow citizens mine rest final success failure course since country founded generation americans summone d give testimony national loyalty graves young americans answered call service surround g lobe trumpet summons call bear arms though arms need call battle though embattled call be ar burden long twilight struggle year year rejoicing hope patient tribulation struggle co mmon enemies man tyranny poverty disease war forge enemies grand global alliance north so uth east west assure fruitful life mankind join historic effort long history world genera tions granted role defending freedom hour maximum danger shrink responsibility welcome be lieve would exchange places people generation energy faith devotion bring endeavor light country serve glow fire truly light world fellow americans ask country ask country fellow citizens world ask america together freedom man finally whether citizens america citizens world ask high standards strength sacrifice ask good conscience sure reward history final judge deeds go forth lead land love asking blessing help knowing earth gods work must tru ly']

3. Speech of president Nixon without stopwords

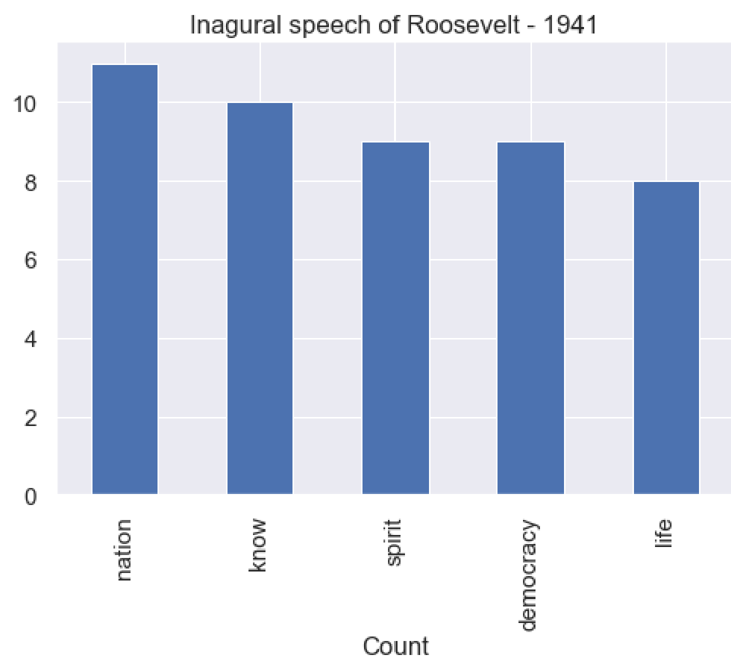
['vice president speaker chief justice senator cook mrs eisenhower fellow citizens great good country share together met four years ago america bleak spirit depressed prospect se emingly endless war abroad destructive conflict home meet today stand threshold new era p eace world central question use peace resolve era enter postwar periods often time retrea t isolation leads stagnation home invites new danger abroad resolve become time great res ponsibilities greatly borne renew spirit promise america enter third century nation past year saw farreaching results new policies peace continuing revitalize traditional friends hips missions peking moscow able establish base new durable pattern relationships among n ations world americas bold initiatives 1972 long remembered year greatest progress since end world war ii toward lasting peace world peace seek world flimsy peace merely interlud e wars peace endure generations come important understand necessity limitations americas role maintaining peace unless america work preserve peace peace unless america work prese rve freedom freedom clearly understand new nature americas role result new policies adopt ed past four years respect treaty commitments support vigorously principle country right impose rule another force continue era negotiation work limitation nuclear arms reduce danger confrontation great powers share defending peace freedom world expect others share time passed america make every nations conflict make every nations future responsibility presume tell people nations manage affairs respect right nation determine future also recognize responsibility nation secure future americas role indispensable preserving worlds peace nations role indispensable preserving peace together rest world resolve move forward beginnings made continue bring walls hostility divided world long build place bridges understanding despite profound differences systems government people world friends build structure peace world weak safe strong respects right live different system would influence others strength ideas force arms accept high responsibility burden gladly gladly chance build peace noblest endeavor nation engage gladly also act greatly meeting responsibilities abroad remain great nation remain great nation act greatly meeting challenges home chance today ever history make life better america ensure better education better health better housing better transportation cleaner environment restore respect law make communities livable insure godgiven right every american full equal opportunity range needs great reach opportunities great bold determination meet needs new ways building structure peace abroad required turning away old policies failed building new era progress home requires turning away old policies failed abroad shift old policies new retreat responsibilities better way peace home shift old policies new retreat responsibilities better way progress abroad home key new responsibilities lies placing division responsibility lived long consequences attempting gather power responsibility washington abroad home time come turn away condescending policies paternalism washington knows best person expected act responsibly responsibility human nature encourage individuals home nations abroad decide locate responsibility places measure others today offer promise purely governmental solution every problem lived long false promise trusting much government asked deliver leads inflated expectations reduced individual effort disappointment frustration erode confidence government people government must learn take less people people remember america built government people welfare work shirking responsibility seeking responsibility lives ask government challenges face together ask government help help national government great vital role play pledge government act act boldly lead boldly important role every one must play individual member community day forward make solemn commitment heart bear responsibility part live ideals together see dawn new age progress america together celebrate 200th anniversary nation proud fulfillment promise world americas longest difficult war comes end learn debate differences civility decency reach one precious quality government cannot provide new level respect rights feelings one another new level respect individual human dignity cherished birthright every american else time come renew faith america recent years faith challenged children taught ashamed country ashamed parents ashamed americas record home role world every turn beset find everything wrong america little right confident judgment history remarkable times privileged live americas record century unparalleled worlds history responsibility generosity creativity progress proud system produced provided freedom abundance widely shared system history world proud four wars engaged century including one bringing end fought selfish advantage help others resist aggression proud bold new initiatives steadfastness peace honor made breakthrough toward creating world world known structure peace last merely time generations come embarking today era presents challenges great nation generation ever faced answer god history conscience way use years stand place hallowed history think others stood think dreams america think recognized needed help far beyond order make dreams come true today ask prayers years ahead may gods help making decisions right america pray help together may worthy challenge pledge together make next four

```
years best four years americas history 200th birthday america young vital began bright be
acon hope world go forward confident hope strong faith one another sustained faith god cr
eated striving always serve purpose']
```

2.3 Which word occurs the most number of times in his inaugural address for each president? Mention the top three words. (after removing the stopwords)

Frequency of first 5 words in 1st speech :

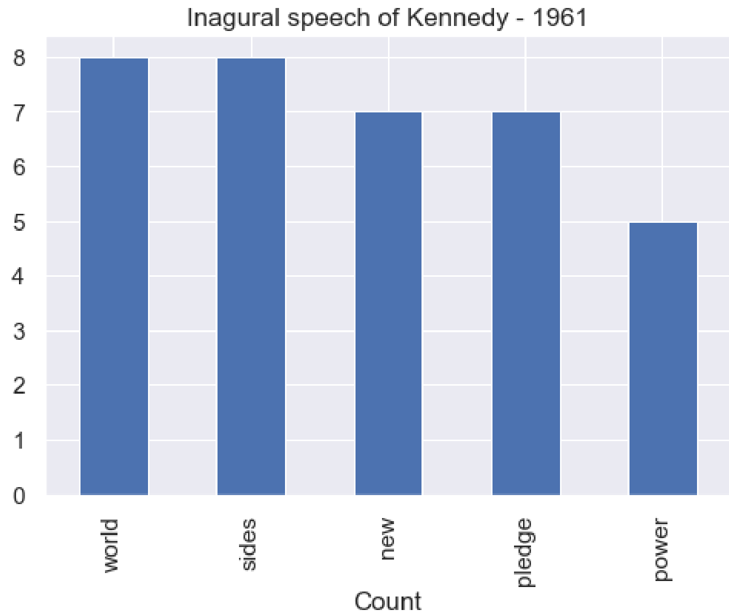
```
nation       11
know         10
spirit        9
democracy     9
life          8
```



2nd speech

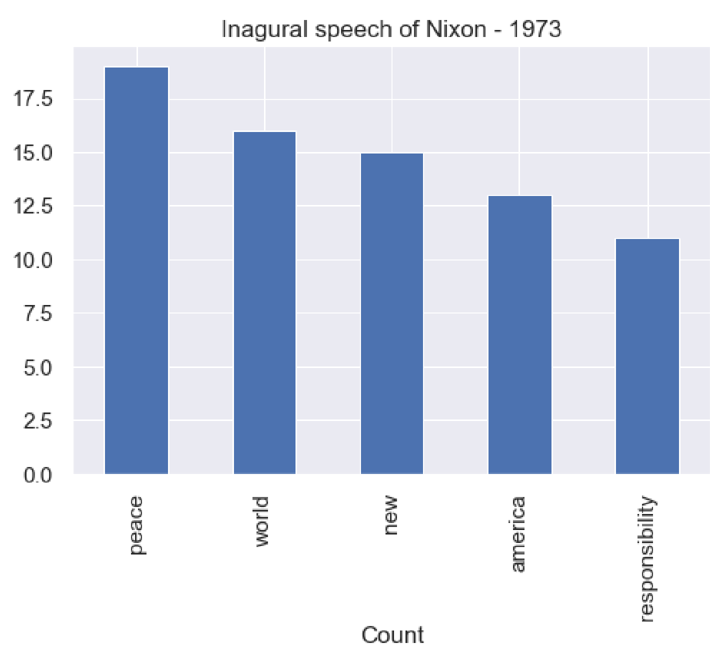
```
world     8
sides     8
new       7
```

```
pledge     7
power      5
```



3rd speech

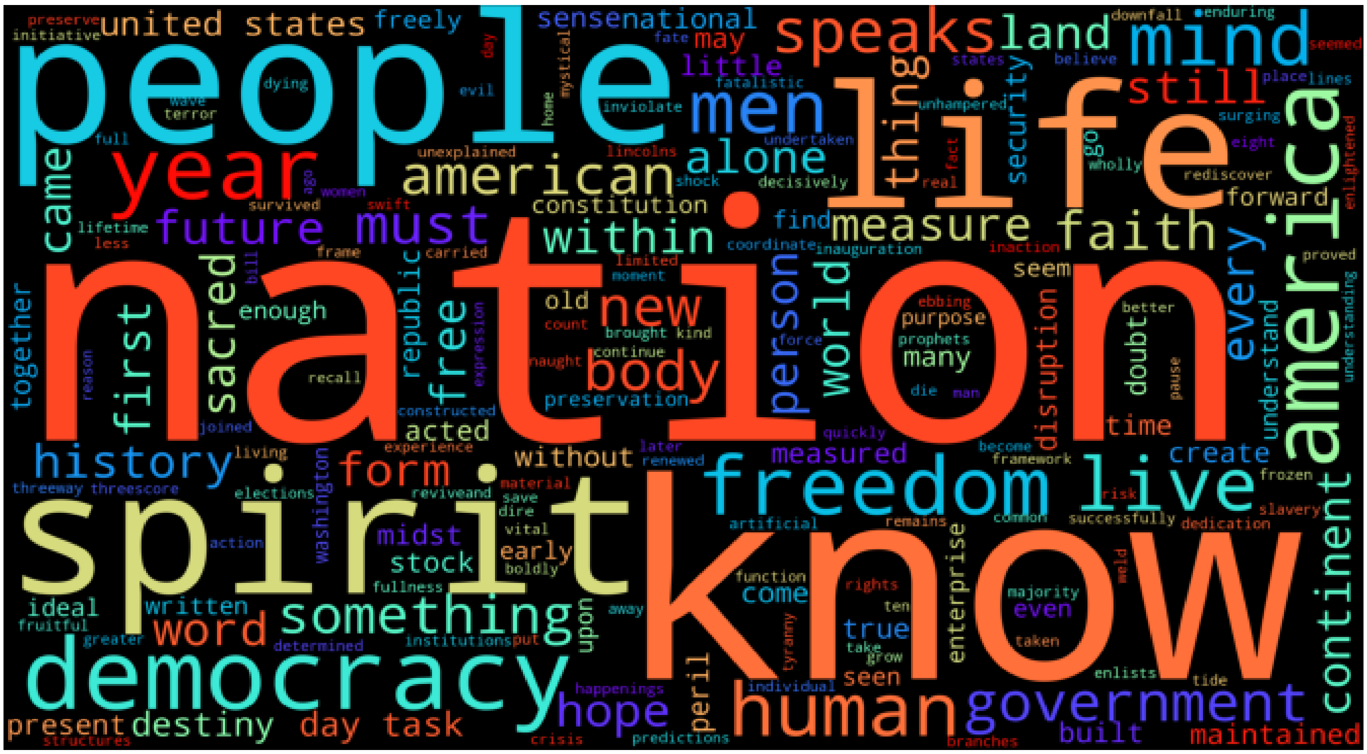
```
peace                  19
world                  16
new                    15
america                13
responsibility         11
```



2.4 Plot the word cloud of each of the speeches of the variable. (after removing the stopwords)

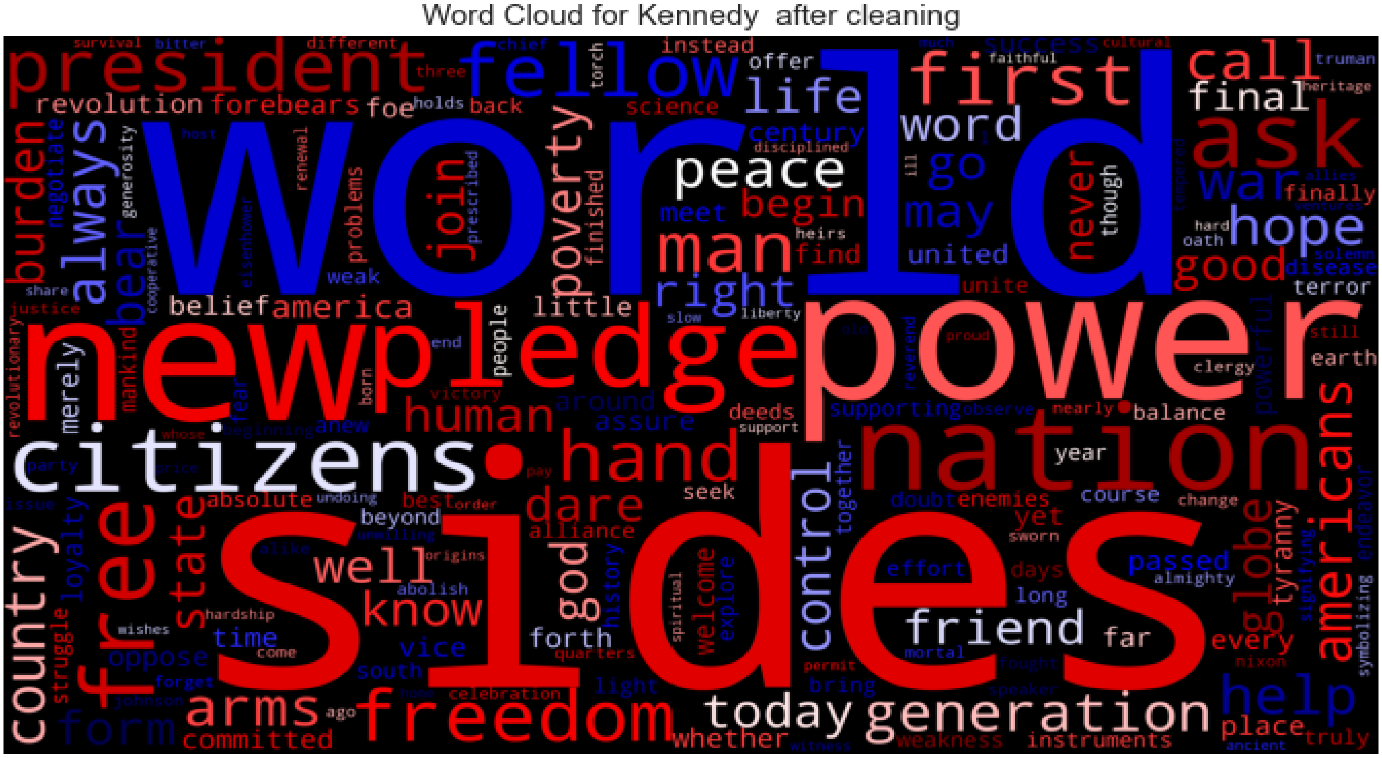
1st Speech

Word Cloud for Roosewelt after cleaning



2nd Speech

Word Cloud for Kennedy after cleaning



3rd Speech

Word Cloud for Nixon after cleaning